This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

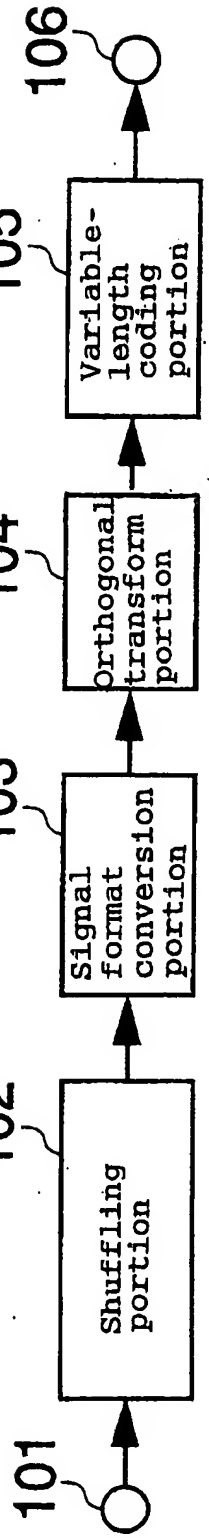

101
102
Shuffling portion
103
Signal format conversion portion
104
Orthogonal transform portion
105
Variable-length coding portion
106

Fig. 1

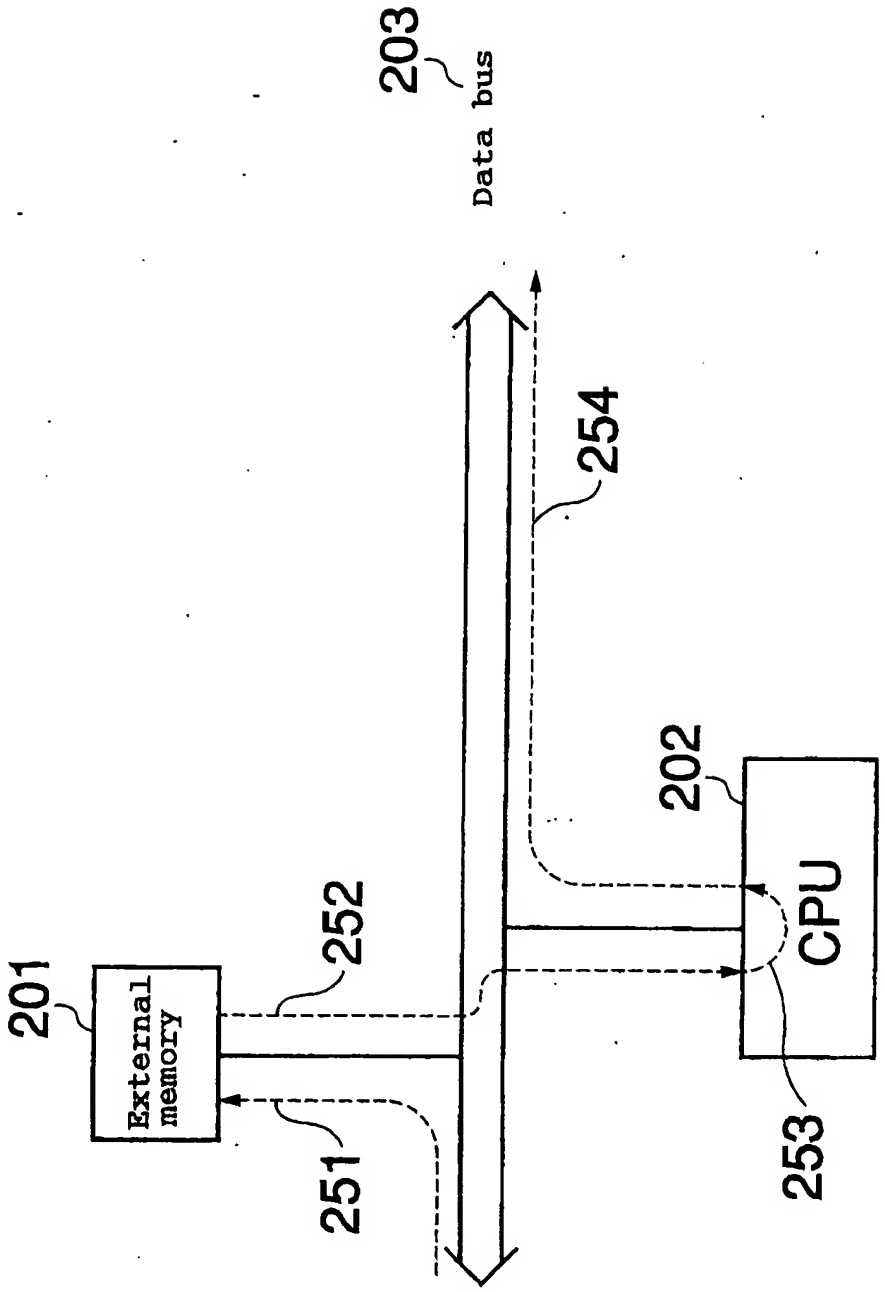


Fig. 2

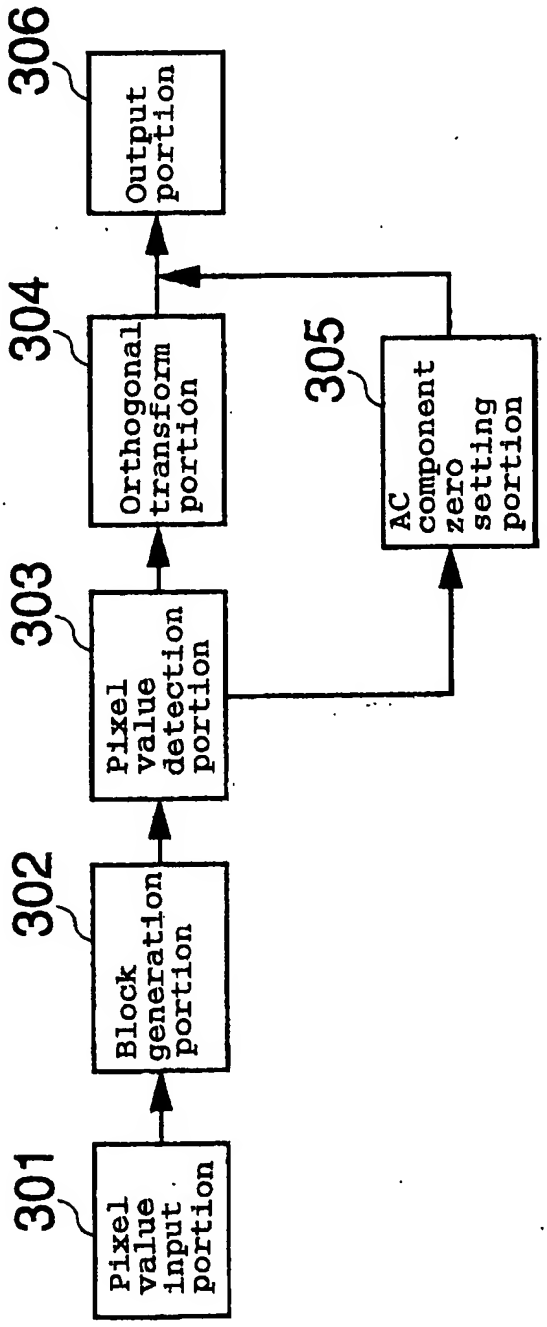

301
Pixel value input portion
302
Block generation portion
303
Pixel value detection portion
304
Orthogonal transform portion
305
AC component zero setting portion
306
Output portion

Fig. 3

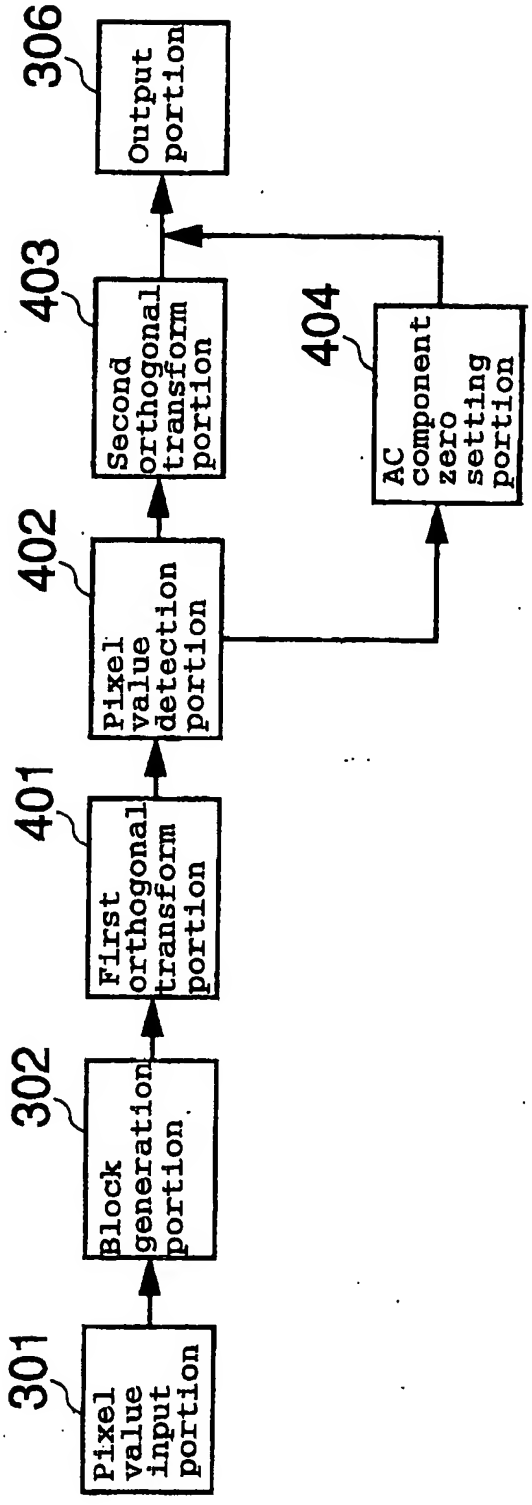

301
Pixel value input portion
302
Block generation portion
401
First orthogonal transform portion
402
Pixel value detection portion
403
Second orthogonal transform portion
404
AC component zero setting portion
306
Output portion

Fig. 4

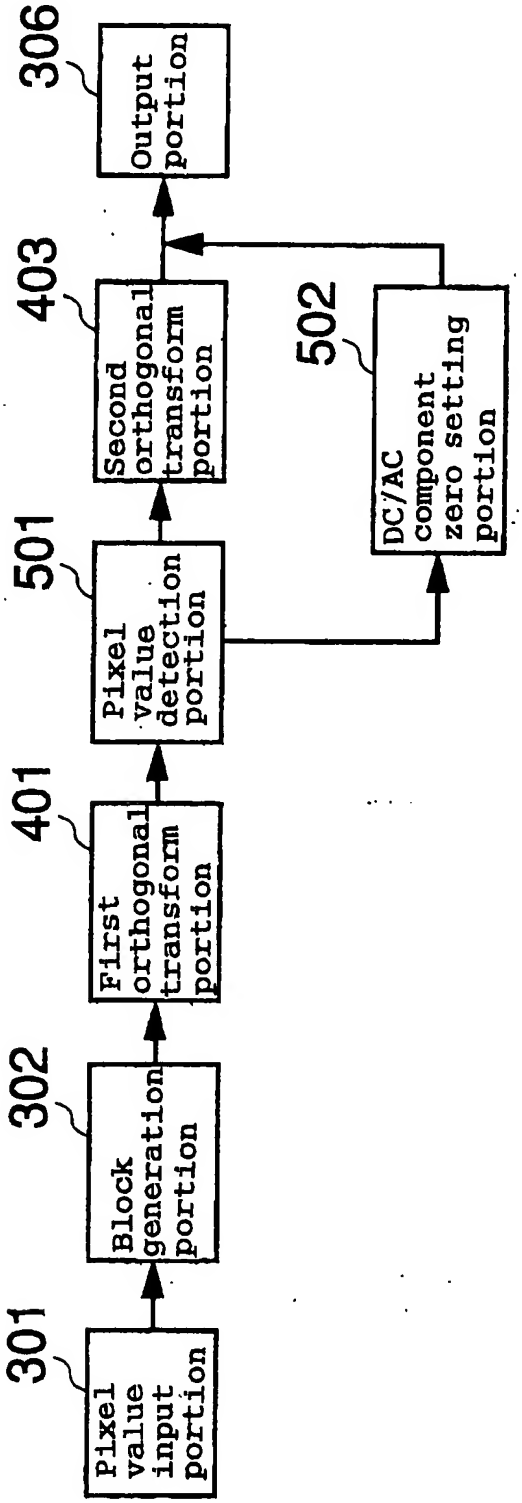

301
Pixel value input portion
302
Block generation portion
401
First orthogonal transform portion
501
Pixel value detection portion
403
Second orthogonal transform portion
306
Output portion
502
DC/AC component zero setting portion

Fig. 5

Fig. 6

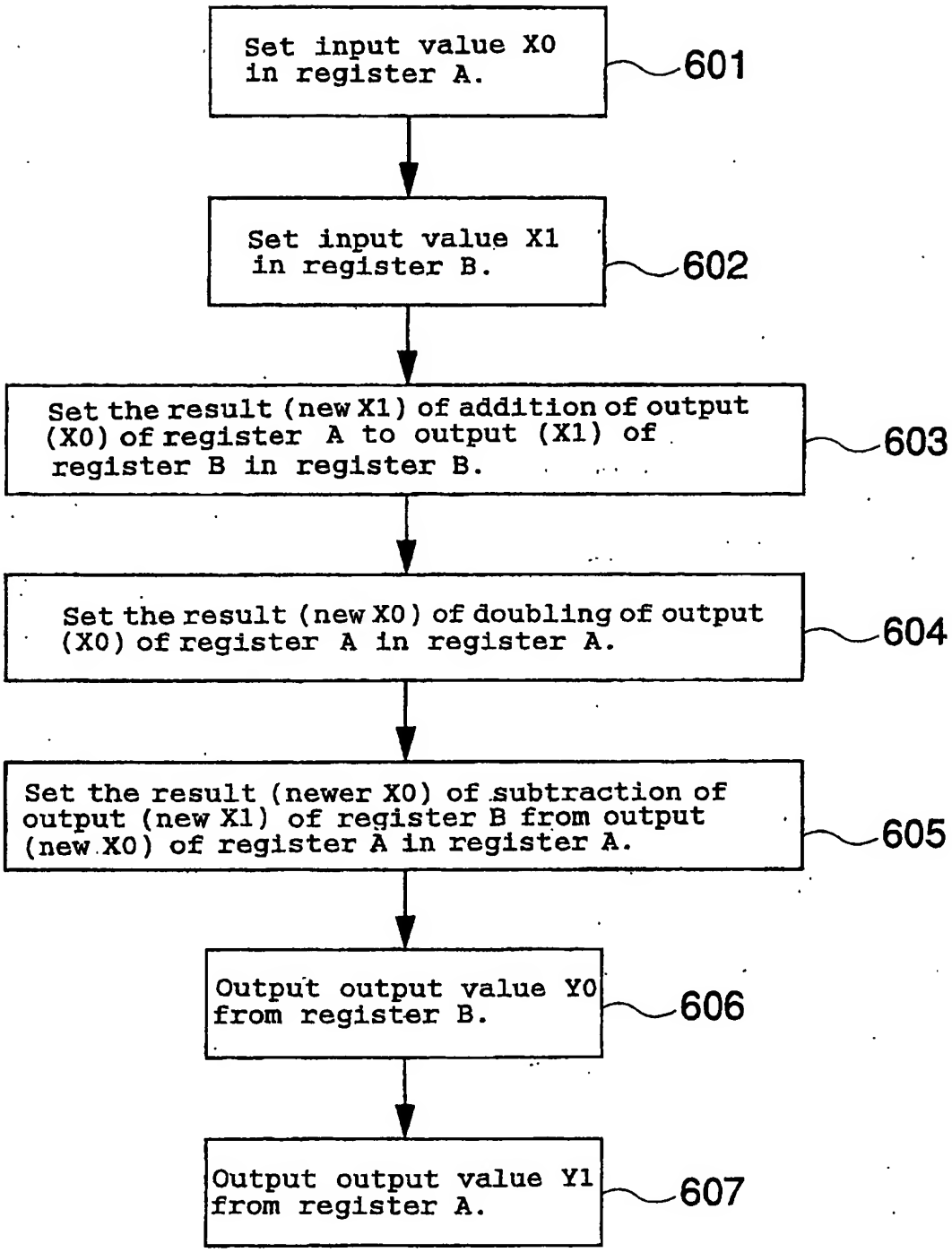

Set input value X0 in register A.
601
Set input value X1 in register B.
602
Set the result (new X1) of addition of output (X0) of register A to output (X1) of register B in register B.
603
Set the result (new X0) of doubling of output (X0) of register A in register A.
604
Set the result (newer X0) of subtraction of output (new X1) of register B from output (new X0) of register A in register A.
605
Output output value Y0 from register B.
606
Output output value Y1 from register A.
607

Fig. 7

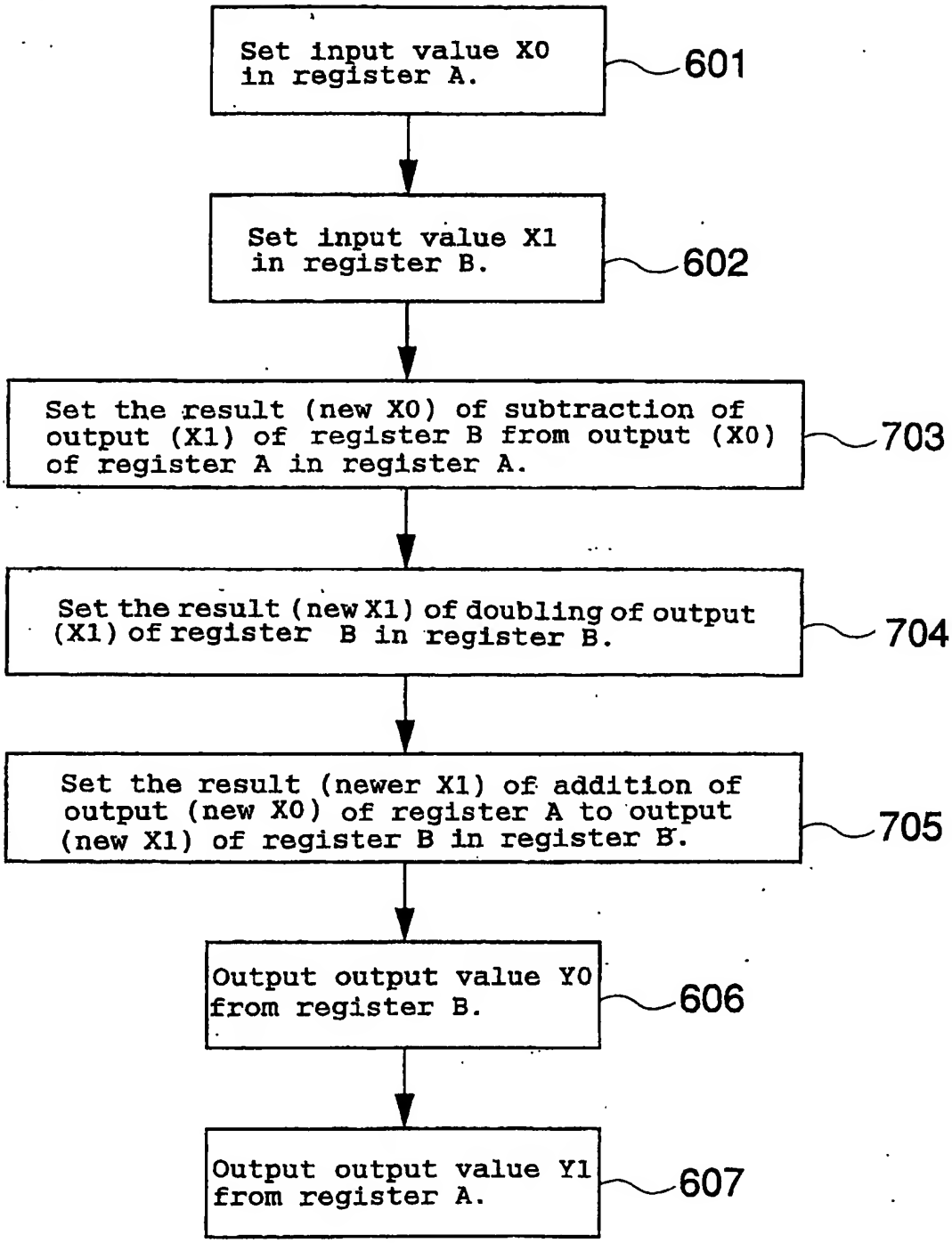

Set input value X0 in register A.
601
Set input value X1 in register B.
602
Set the result (new X0) of subtraction of output (X1) of register B from output (X0) of register A in register A.
703
Set the result (new X1) of doubling of output (X1) of register B in register B.
704
Set the result (newer X1) of addition of output (new X0) of register A to output (new X1) of register B in register B.
705
Output output value Y0 from register B.
606
Output output value Y1 from register A.
607

Fig. 8

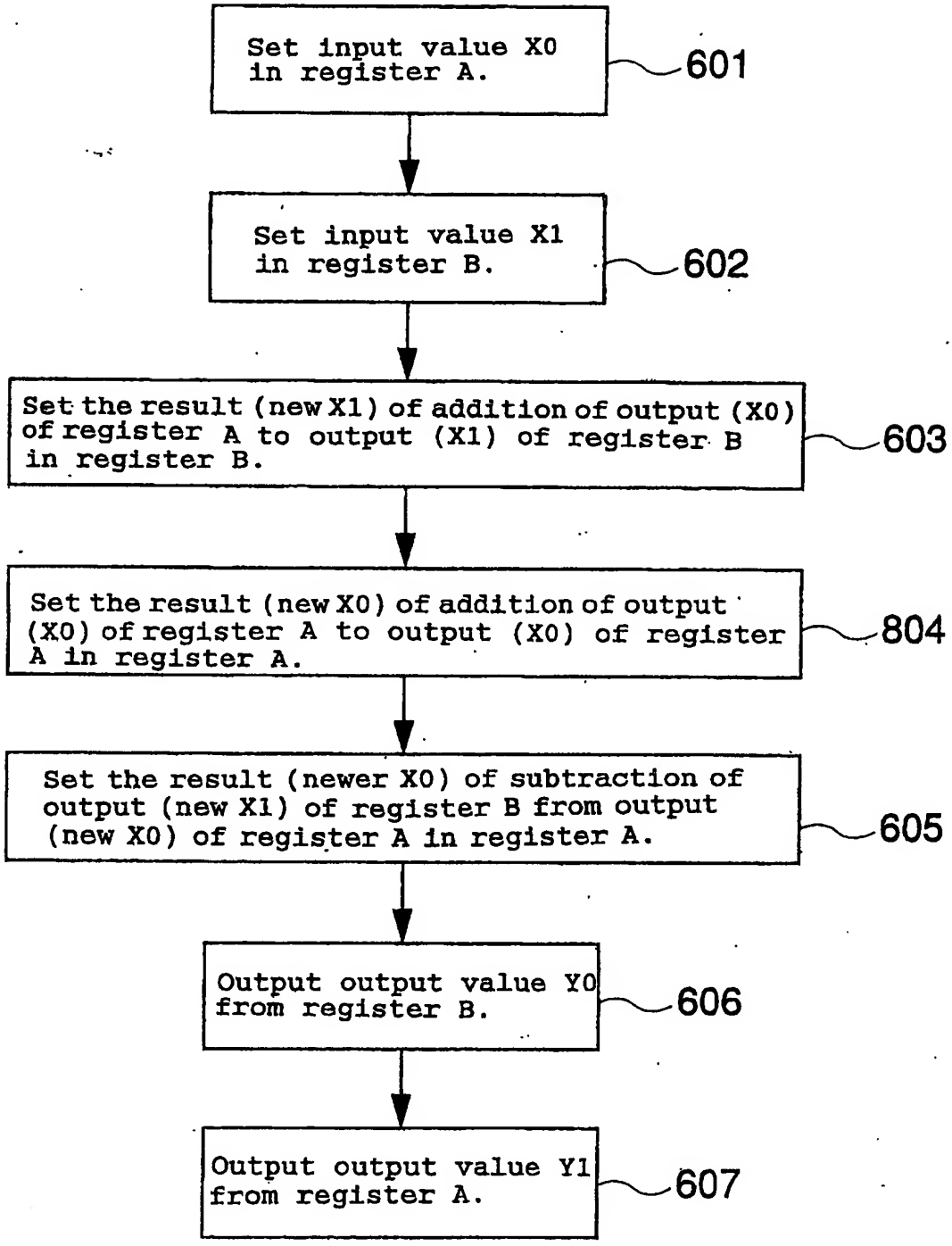

Set input value X0 in register A.
601
Set input value X1 in register B.
602
Set the result (new X1) of addition of output (X0) of register A to output (X1) of register B in register B.
603
Set the result (new X0) of addition of output (X0) of register A to output (X0) of register A in register A.
804
Set the result (newer X0) of subtraction of output (new X1) of register B from output (new X0) of register A in register A.
605
Output output value Y0 from register B.
606
Output output value Y1 from register A.
607

Fig. 9

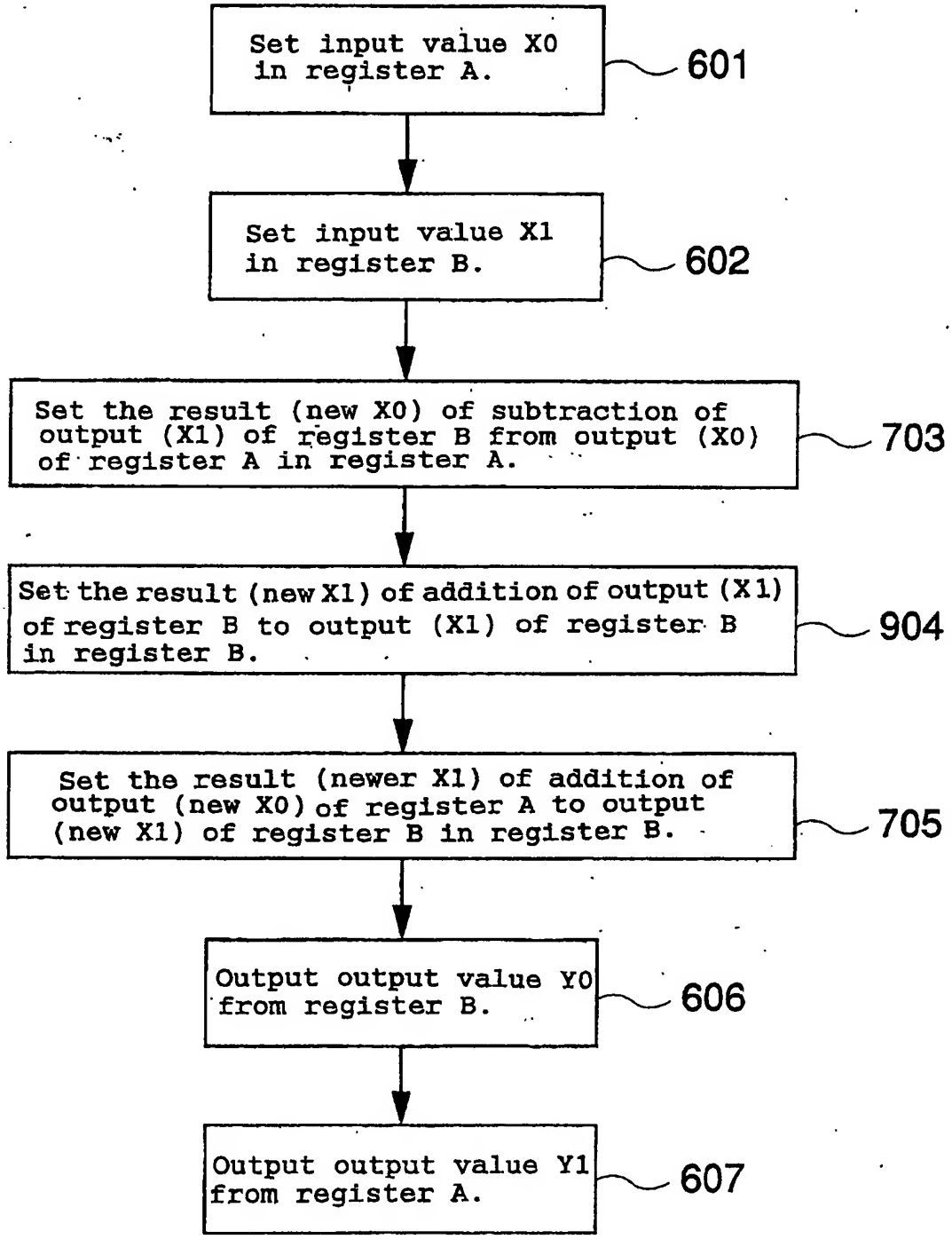

Set input value X0 in register A.
601
Set input value X1 in register B.
602
Set the result (new X0) of subtraction of output (X1) of register B from output (X0) of register A in register A.
703
Set the result (new X1) of addition of output (X1) of register B to output (X1) of register B in register B.
904
Set the result (newer X1) of addition of output (new X0) of register A to output (new X1) of register B in register B.
705
Output output value Y0 from register B.
606
Output output value Y1 from register A.
607

Fig. 10

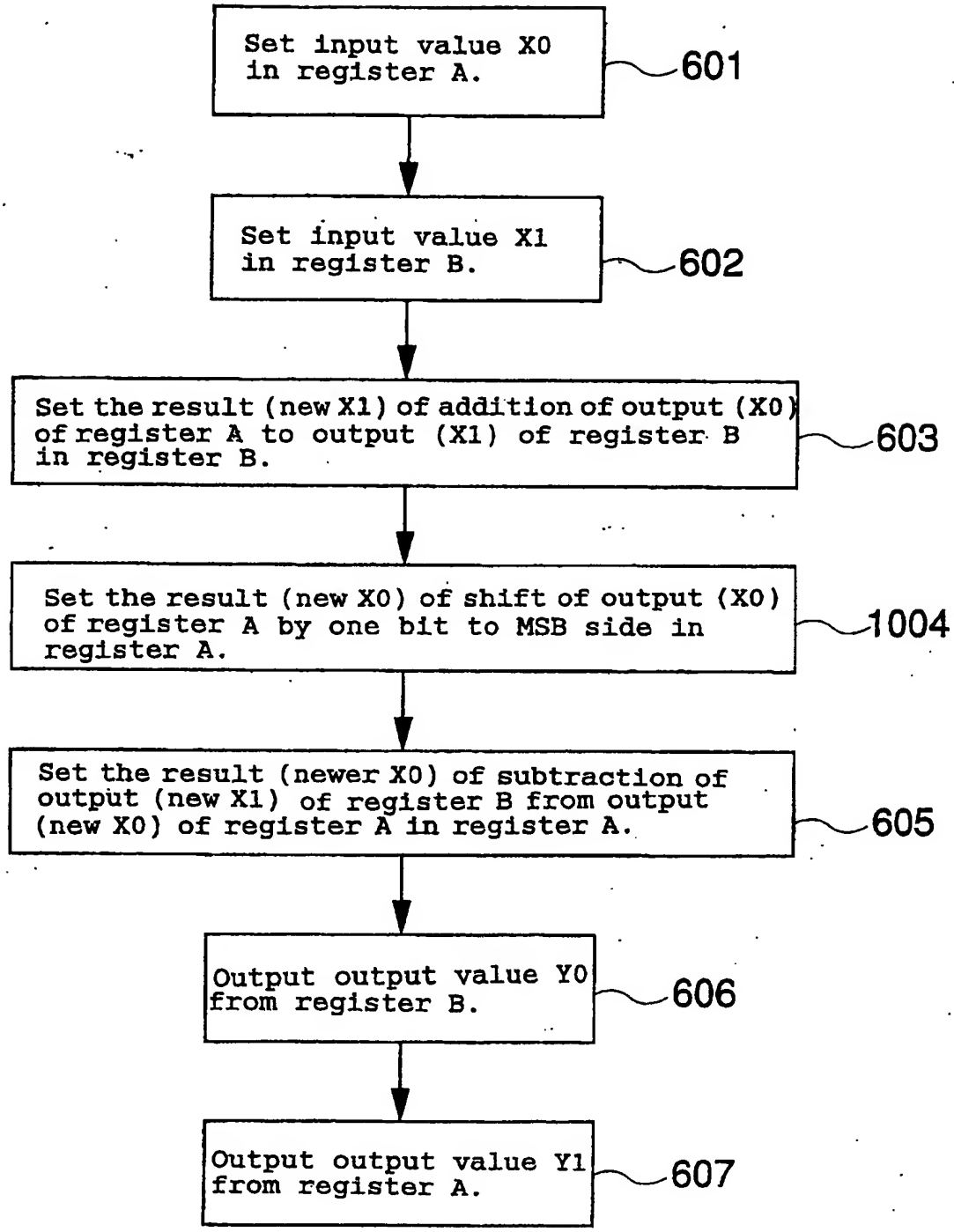

Set input value X0 in register A.
601
Set input value X1 in register B.
602
Set the result (new X1) of addition of output (X0) of register A to output (X1) of register B in register B.
603
Set the result (new X0) of shift of output (X0) of register A by one bit to MSB side in register A.
1004
Set the result (newer X0) of subtraction of output (new X1) of register B from output (new X0) of register A in register A.
605
Output output value Y0 from register B.
606
Output output value Y1 from register A.
607

Fig. 11

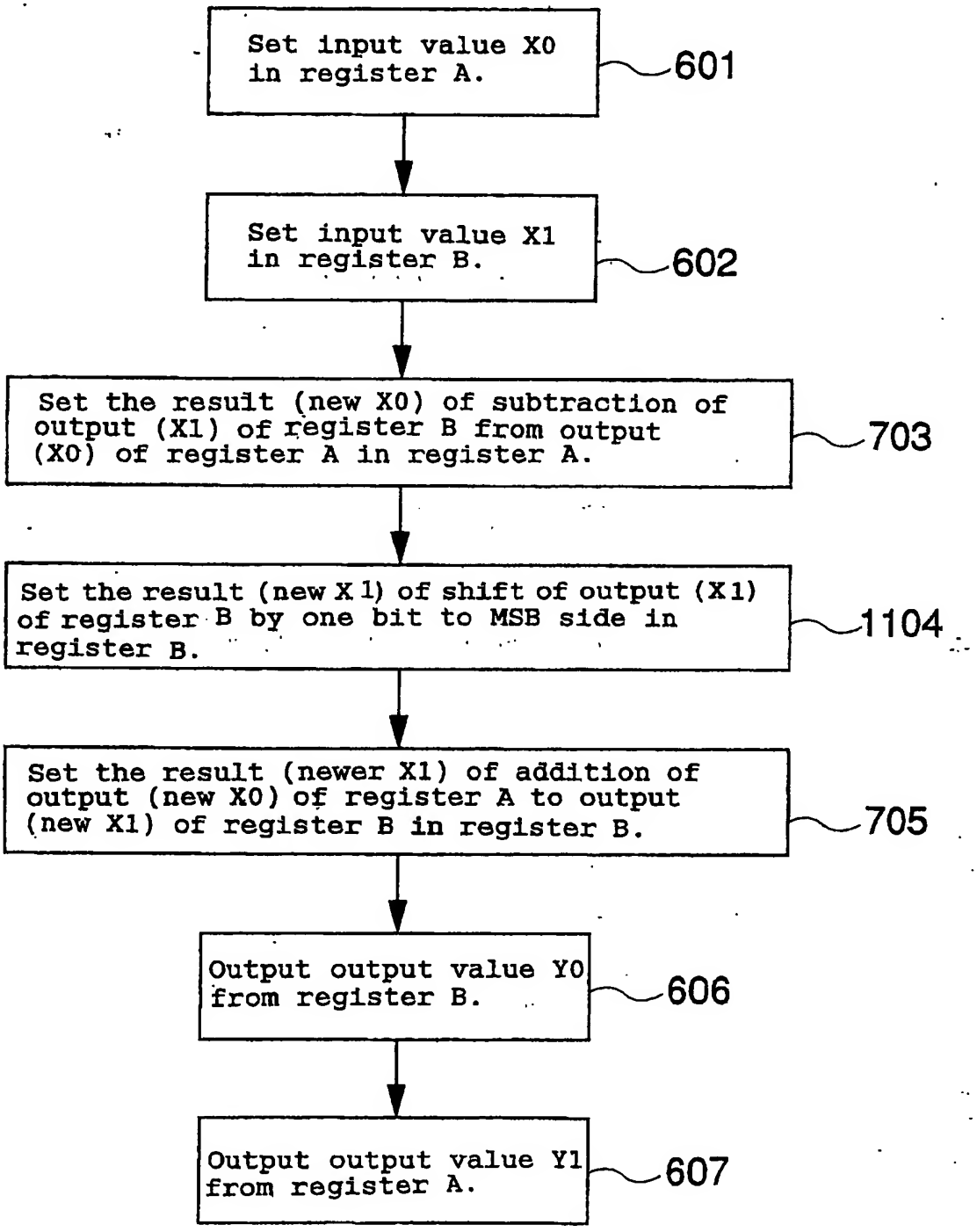

Set input value X0 in register A.
601
Set input value X1 in register B.
602
Set the result (new X0) of subtraction of output (X1) of register B from output (X0) of register A in register A.
703
Set the result (new X1) of shift of output (X1) of register B by one bit to MSB side in register B.
1104
Set the result (newer X1) of addition of output (new X0) of register A to output (new X1) of register B in register B.
705
Output output value Y0 from register B.
606
Output output value Y1 from register A.
607

Fig. 12

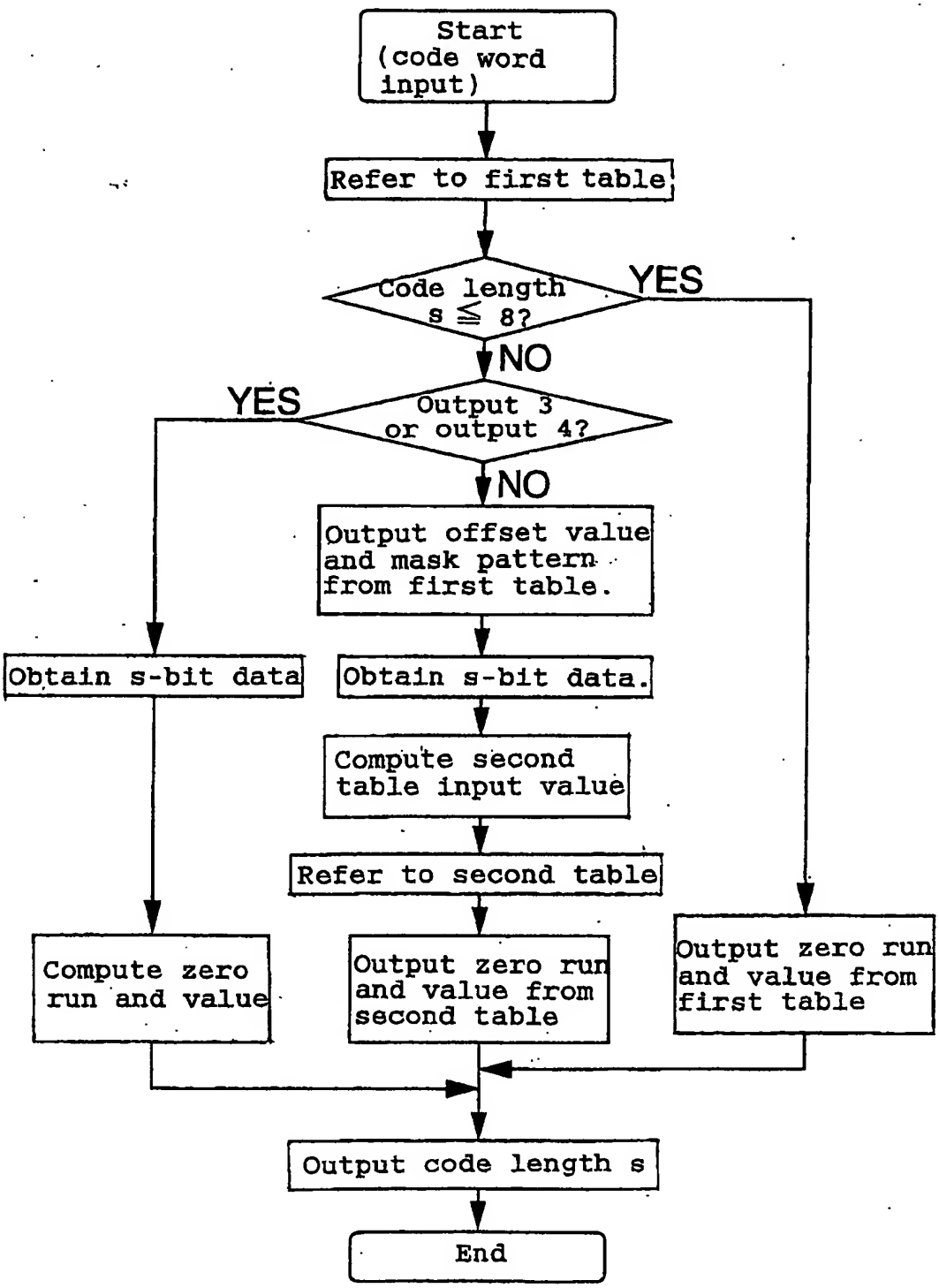

Start (code word input)
Refer to first table
Code length s ≦ 8?
YES
NO
Output 3 or output 4?
YES
NO
Output offset value and mask pattern from first table.
Obtain s-bit data
Obtain s-bit data.
Compute second table input value
Refer to second table
Compute zero run and value
Output zero run and value from second table
Output zero run and value from first table
Output code length s
End

Fig. 13

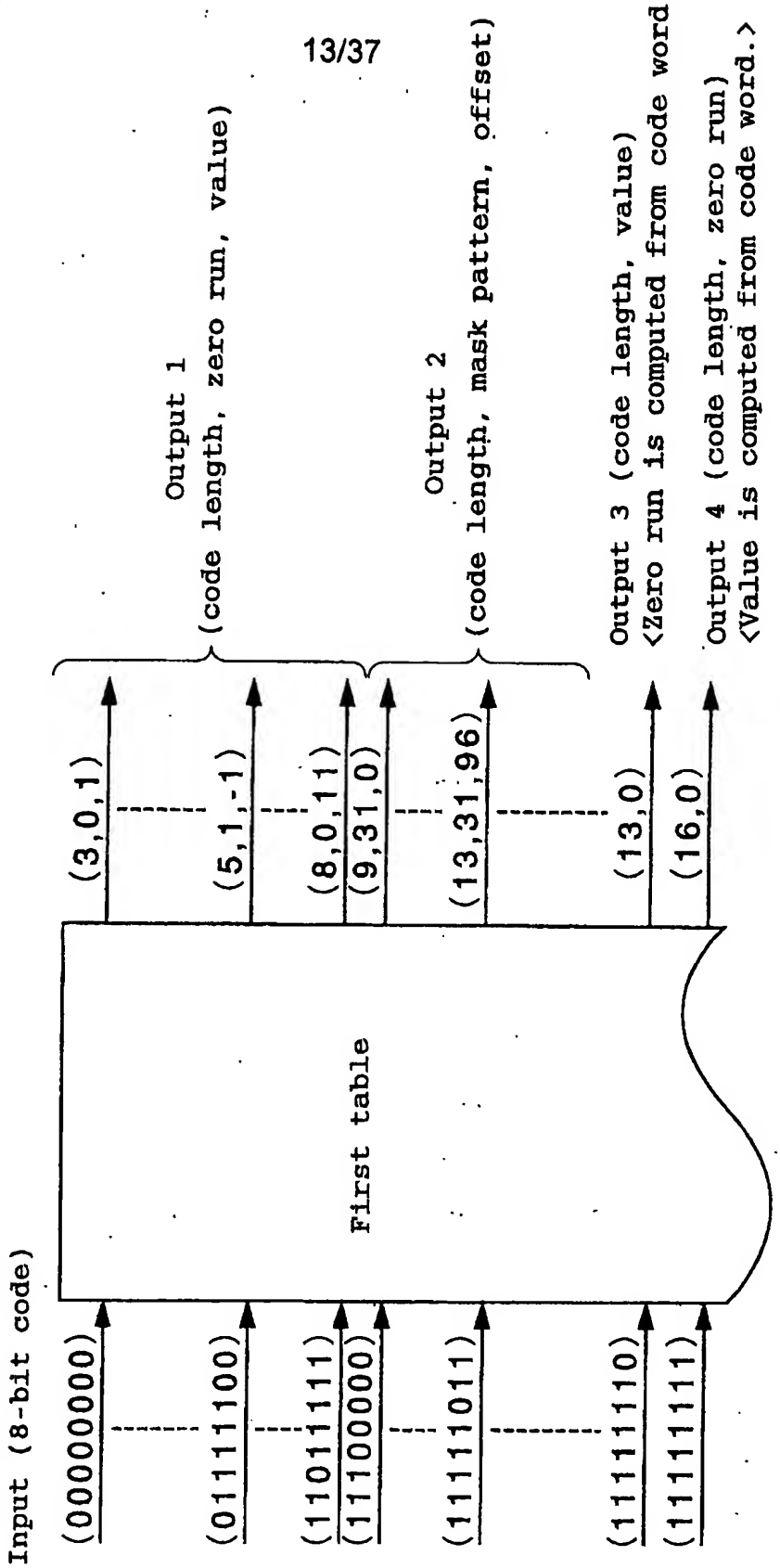

Input (8-bit code)
(00000000)
(01111100)
(11011111)
(11100000)
(11111011)
(11111110)
(11111111)
First table
(3,0,1)
(5,1,-1)
(8,0,11)
(9,31,0)
(13,31,96)
(13,0)
(16,0)
Output 1
(code length, zero run, value)
Output 2
(code length, mask pattern, offset)
Output 3 (code length, value)
<Zero run is computed from code word.>
Output 4 (code length, zero run)
<Value is computed from code word.>

Fig. 14

First table output		
Code length	Mask pattern	Offset value (f)
9	11111	0
10	11111	32
11	1111	64
12	1111	80
13	11111	96

Code word to be obtained	
	Masked value (f)
1110	00000
1110	00001
:	:
1110	11110
1110	11111
11110	00000
11110	00001
:	:
11110	11110
11110	11111
1111100	0000
:	:
1111100	1111
11111010	0000
:	:
11111010	1111
11111011	00000
	00001
:	:
11111011	11110
11111011	11111

Second table		
Input address (f + t)	Zero run	Value
0	7	1
1	7	-1
:	:	:
30	0	17
31	0	-17
32	11	1
33	11	-1
:	:	:
62	0	22
63	0	-22
64	5	3
:	:	:
79	1	0
80	6	3
:	:	:
95	5	0
96	7	2
97	7	-2
:	:	:
126	1	17
127	1	-17

Fig. 15

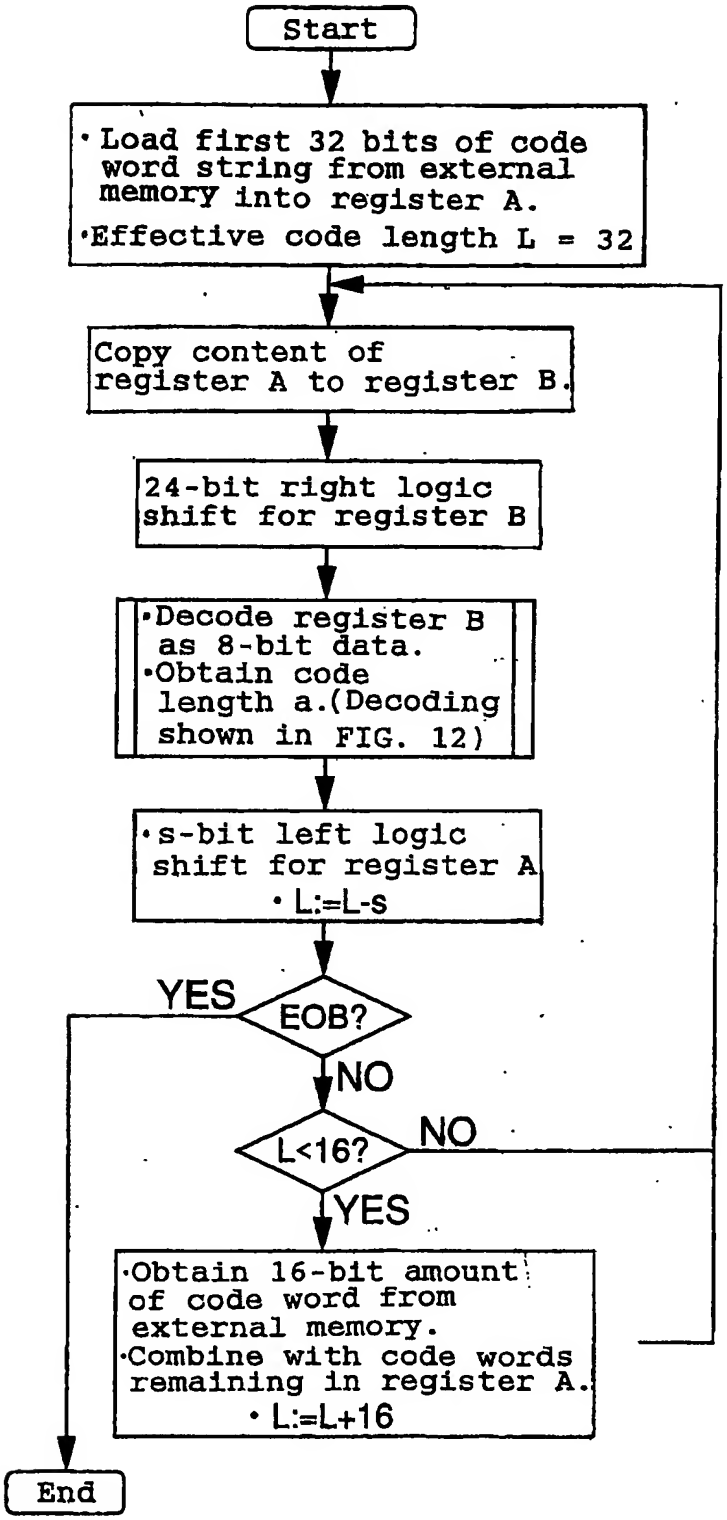

Start
• Load first 32 bits of code word string from external memory into register A.
•Effective code length L = 32
Copy content of register A to register B.
24-bit right logic shift for register B
•Decode register B as 8-bit data.
•Obtain code length a.(Decoding shown in FIG. 12)
•s-bit left logic shift for register A
• L:=L-s
YES
EOB?
NO
L<16?
NO
YES
•Obtain 16-bit amount of code word from external memory.
•Combine with code words remaining in register A.
• L:=L+16
End

Fig. 16

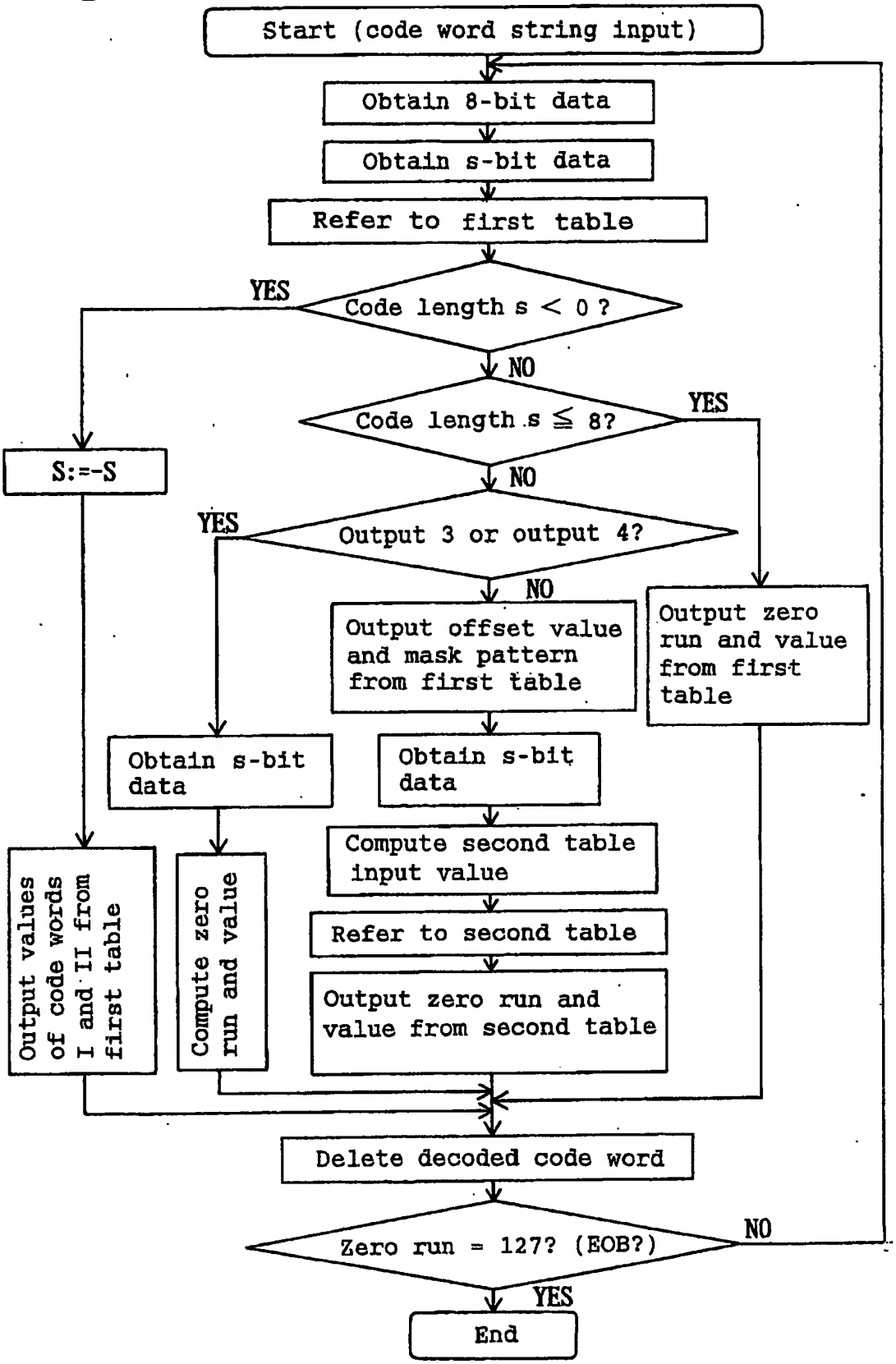

Start (code word string input)
Obtain 8-bit data
Obtain s-bit data
Refer to first table
Code length s < 0 ?
YES
NO
Code length s ≦ 8?
YES
NO
S:=-S
Output 3 or output 4?
YES
NO
Output offset value and mask pattern from first table
Output zero run and value from first table
Obtain s-bit data
Obtain s-bit data
Compute second table input value
Refer to second table
Output zero run and value from second table
Output values of code words I and II from first table
Compute zero run and value
Delete decoded code word
Zero run = 127? (EOB?)
NO
YES
End

Fig. 17

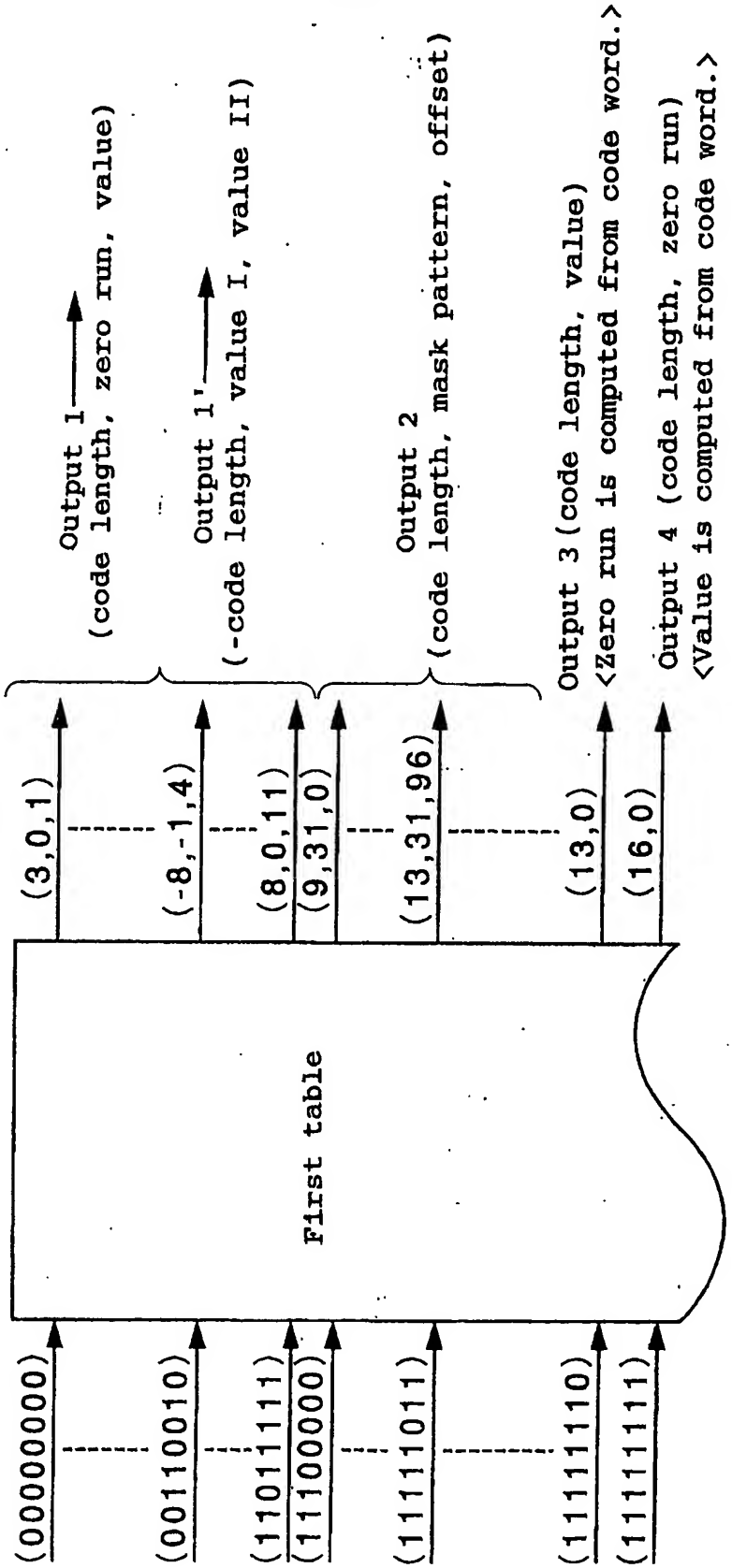
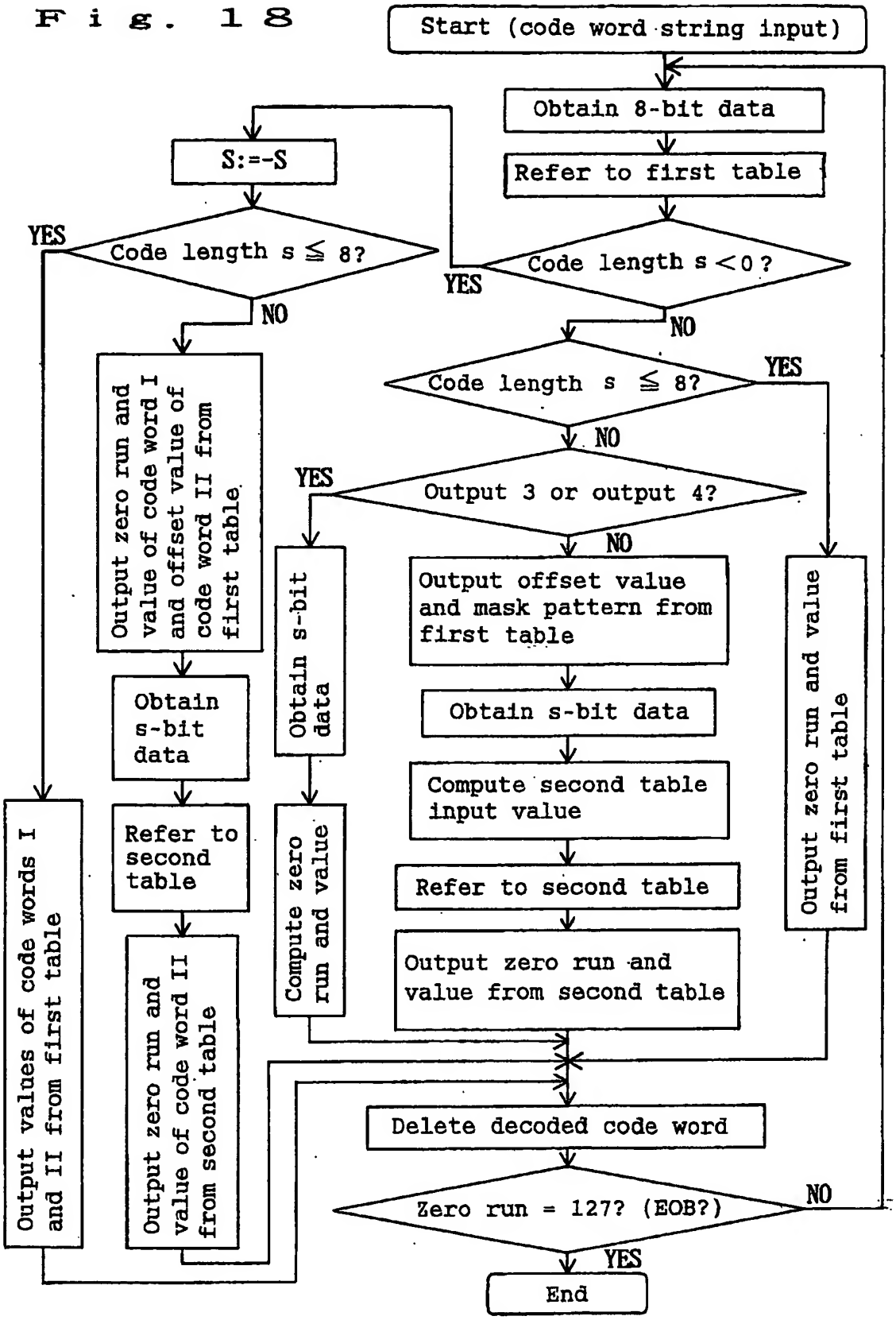

Input
(8-bit code)
(00000000)
(00110010)
(11011111)
(11100000)
(11111011)
(11111110)
(11111111)
First table
(3,0,1)
(-8,-1,4)
(8,0,11)
(9,31,0)
(13,31,96)
(13,0)
(16,0)
Output 1
(code length, zero run, value)
Output 1'
(-code length, value I, value II)
Output 2
(code length, mask pattern, offset)
Output 3 (code length, value)
<Zero run is computed from code word.>
Output 4 (code length, zero run)
<Value is computed from code word.>

Fig. 18


Start (code word string input)
Obtain 8-bit data
Refer to first table
Code length s < 0 ?
YES
NO
S:=-S
Code length s ≦ 8?
YES
NO
Output zero run and value of code word I and offset value of code word II from first table
Obtain s-bit data
Refer to second table
Output zero run and value of code word II from second table
Output values of code words I and II from first table
Code length s ≦ 8?
YES
NO
Output 3 or output 4?
YES
NO
Obtain s-bit data
Compute zero run and value
Output offset value and mask pattern from first table
Obtain s-bit data
Compute second table input value
Refer to second table
Output zero run and value from second table
Output zero run and value from first table
Delete decoded code word
Zero run = 127? (EOB?)
NO
YES
End

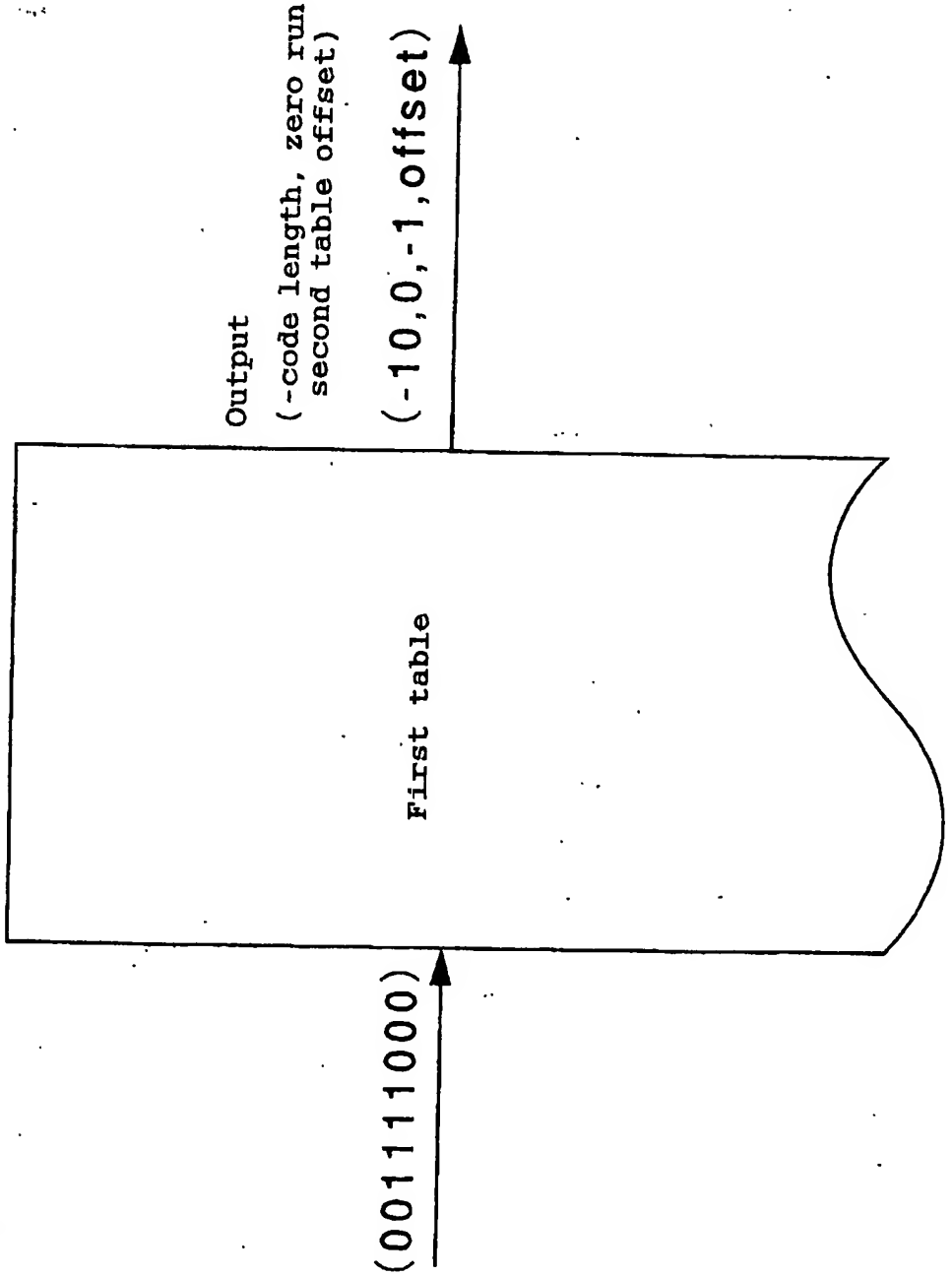

(001111000)
First table
(-10,0,-1,offset)
Output
(-code length, zero run, value,
second table offset)

Fig. 19

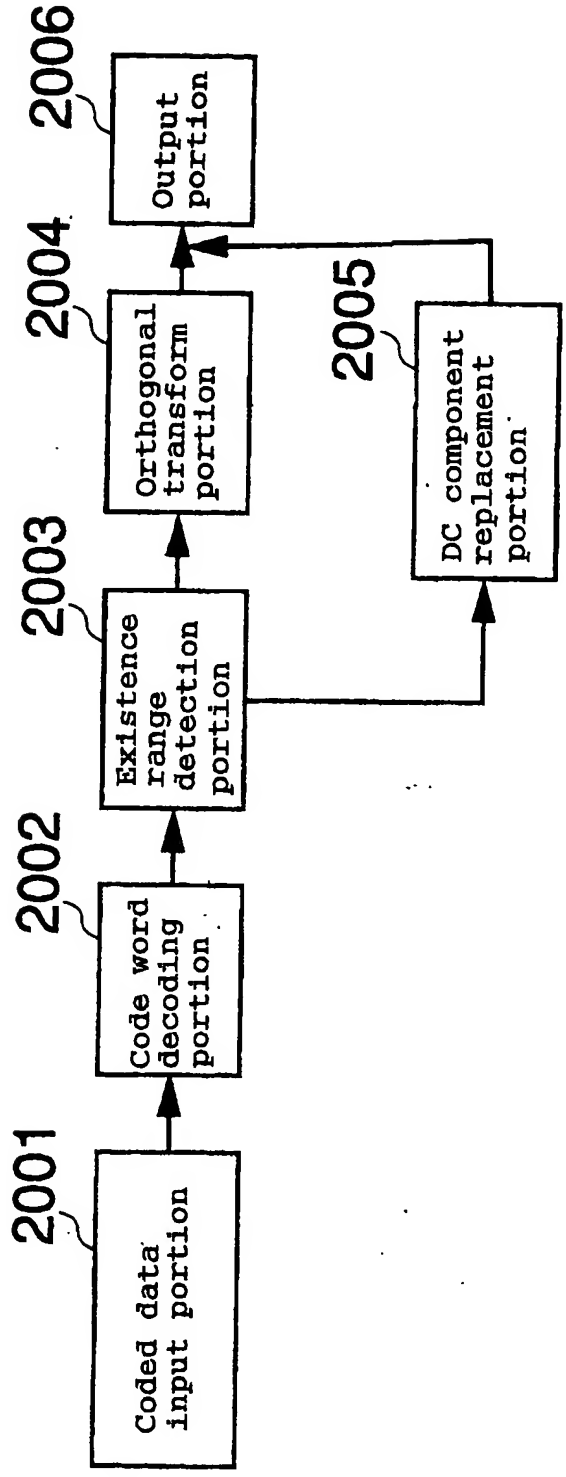

2001
Coded data input portion
2002
Code word decoding portion
2003
Existence range detection portion
2004
Orthogonal transform portion
2005
DC component replacement portion
2006
Output portion

Fig. 20

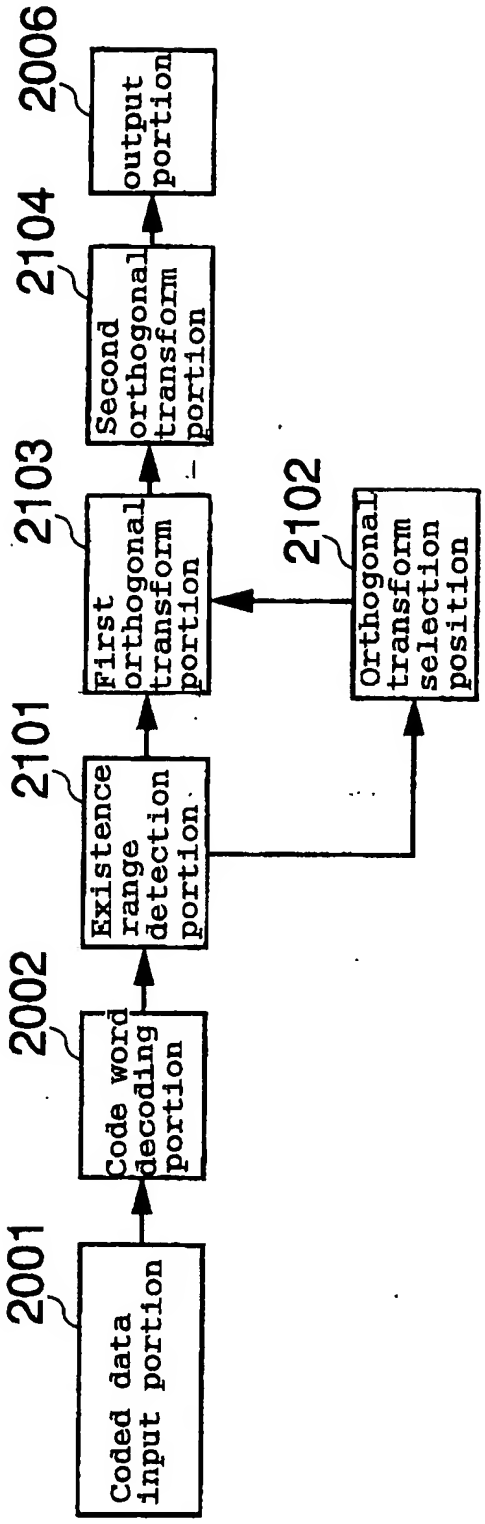

2001
Coded data input portion
2002
Code word decoding portion
2101
Existence range detection portion
2102
Orthogonal transform selection position
2103
First orthogonal transform portion
2104
Second orthogonal transform portion
2006
output portion

Fig. 21

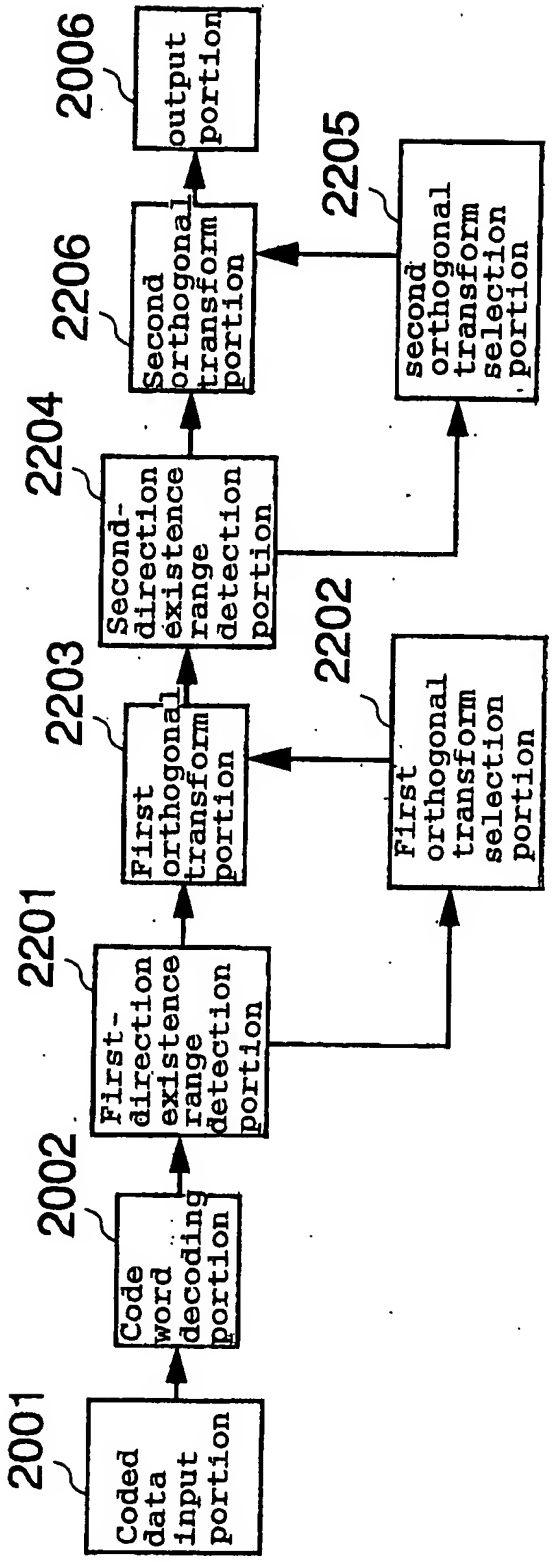

2001
Coded data input portion
2002
Code word decoding portion
2201
First-direction existence range detection portion
2203
First orthogonal transform portion
2204
Second-direction existence range detection portion
2206
Second orthogonal transform portion
2006
output portion
2202
First orthogonal transform selection portion
2205
second orthogonal transform selection portion

Fig. 22

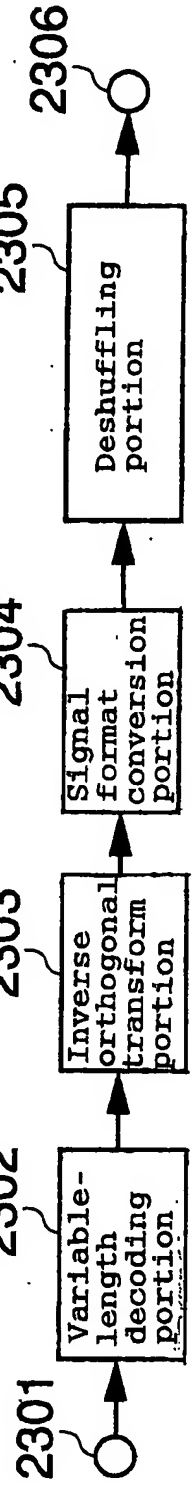

2306
2305
Deshuffling portion
2304
Signal format conversion portion
2303
Inverse orthogonal transform portion
2302
Variable-length decoding portion
2301

Fig. 23

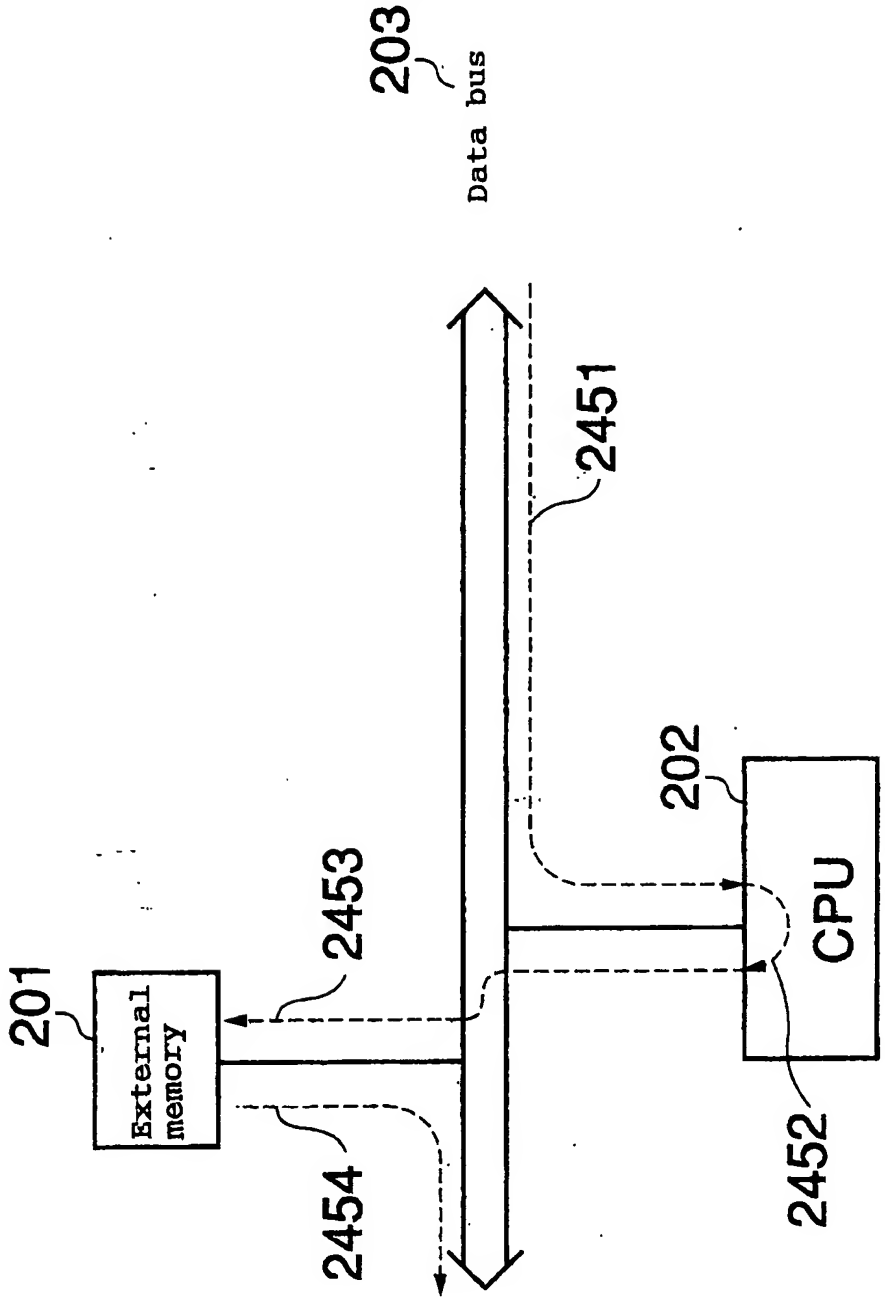


Fig. 24

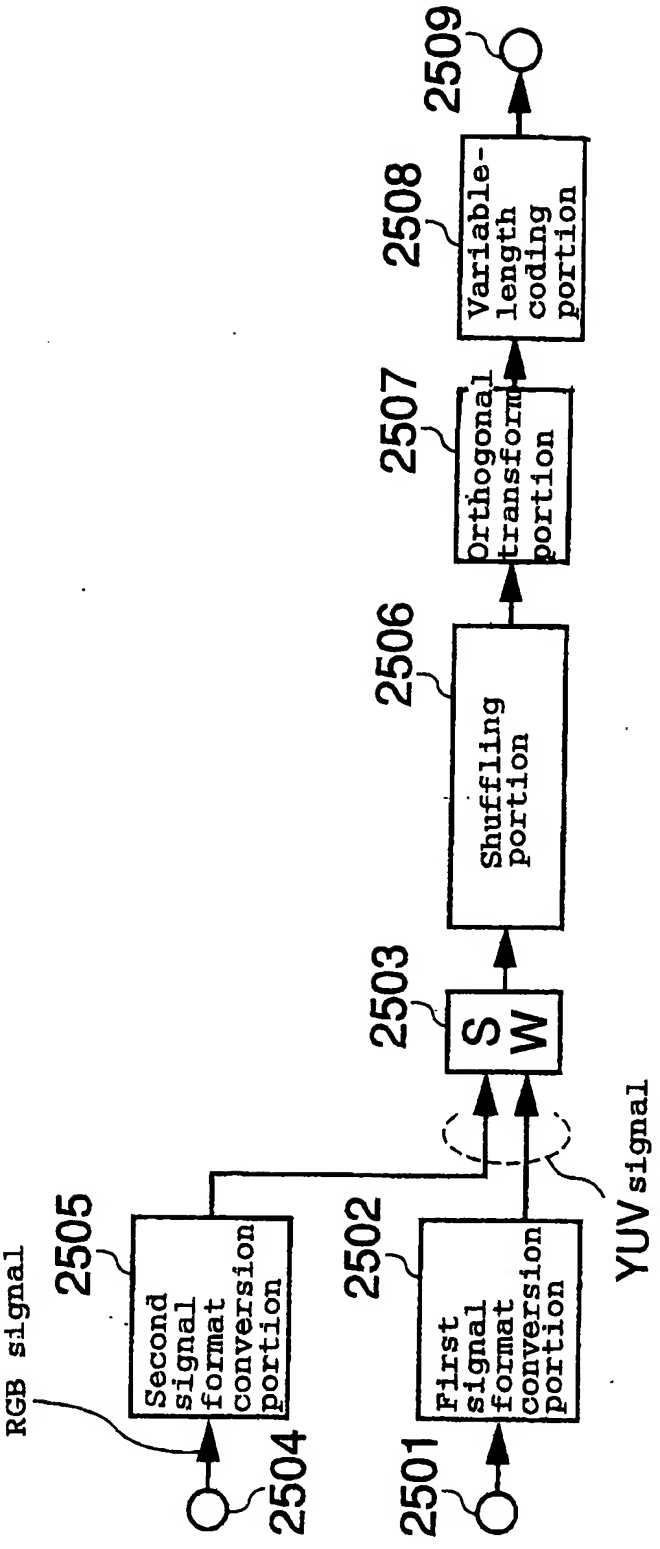

2509
2508
Variable-length coding portion
2507
Orthogonal transform portion
2506
Shuffling portion
2503
S W
2505
Second signal format conversion portion
2502
First signal format conversion portion
RGB signal
YUV signal
2504
2501

Fig. 25

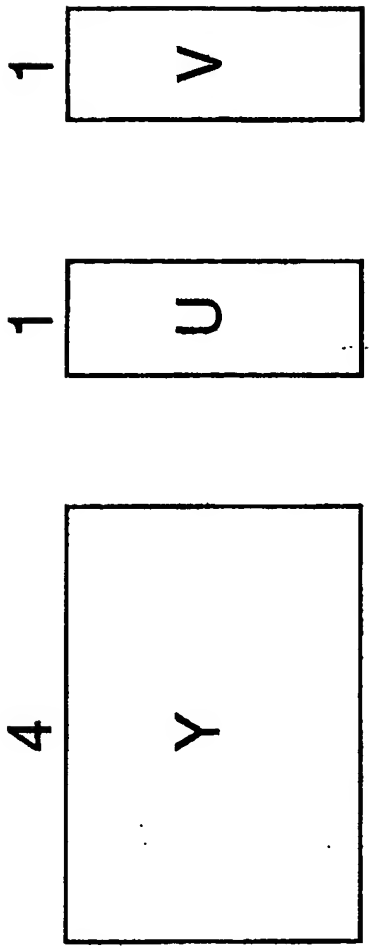


Fig. 26

Fig. 27

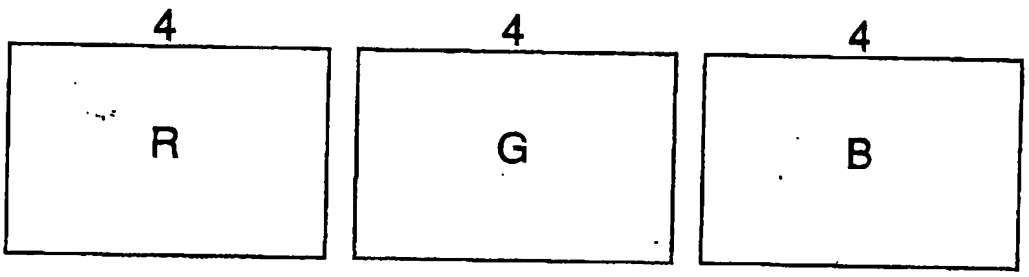
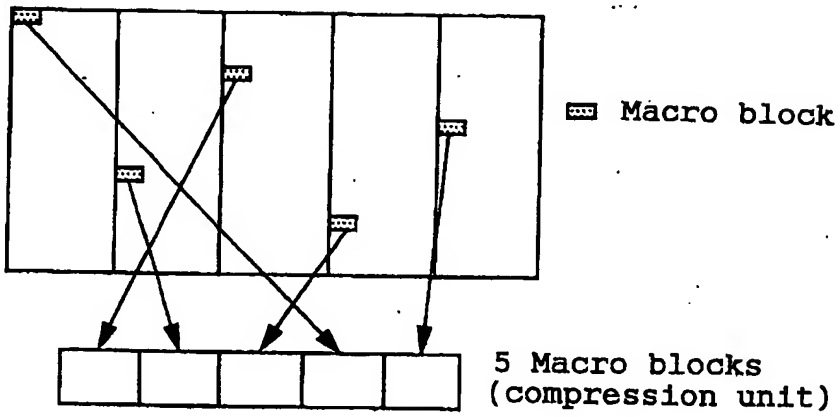


Fig. 28



(Zero run, value)		Code	Code length
0	1	00s	2+1
0	2	010s	3+1
EOB		0110	4
1	1	0111s	4+1
0	3	1000s	
0	4	1001s	
2	1	10100s	5+1
1	2	10101s	
0	5	10110s	
0	6	10111s	
3	1	110000s	6+1
4	1	110001s	
0	7	110010s	
0	8	110011s	
5	1	11010005	7+1
6	1	11010015	
2	2	11010105	
1	3	11010115	
1	4	11011005	
0	9	11011015	
0	10	11011105	
0	11	11011115	
7	1	111000005	8+1
8	1	111000015	
9	1	111000105	
10	1	111000115	
3	2	111001005	
4	2	111001015	
2	3	111001105	
1	5	111001115	
1	6	111010005	
1	7	111010015	
0	12	111010105	
0	13	111010115	
0	14	111011005	
0	15	111011015	
0	16	111011105	
0	17	111011115	

(Zero run, value)		Code	Code length
11	1	1111000005	9+1
12	1	1111000015	
13	1	1111000105	
14	1	1111000115	
5	2	1111001005	
6	2	1111001015	
3	3	1111001105	
4	3	1111001115	
2	4	1111010005	
2	5	1111010015	
1	8	1111010105	
0	18	1111010115	
0	19	1111011005	
0	20	1111011015	
0	21	1111011105	
0	22	1111011115	
5	3	11111000005	10+1
3	4	11111000015	
3	5	11111000105	
2	6	11111000115	
1	9	11111001005	
1	10	11111001015	
1	11	11111001105	
0	0	11111001110	11
1	0	11111001111	
6	3	11111010000s	11+1
4	4	11111010001s	
3	6	11111010010s	
1	12	11111010011s	
1	13	11111010100s	
1	14	11111010101s	
2	0	11111010110	12
3	0	11111010111	
4	0	11111011000	
5	0	11111011001	

	(Zero run, value)		Code		Code length
	7	2	1111110110000s		12+1
	8	2	1111110110001s		
	9	2	1111110110010s		
	10	2	1111110110011s		
	7	3	1111110110100s		
	8	3	1111110110101s		
	4	5	1111110110110s		
	3	7	1111110110111s		
	2	7	1111110111000s		
	2	8	1111110111001s		
	2	9	1111110111010s		
	2	10	1111110111011s		
	2	11	1111110111100s		
	1	15	1111110111101s		
	1	16	1111110111110s		
	1	17	1111110111111s		
Output 3	6	0	1111111000011		13
	7	0	1111111000011		
	R	0	1111110	Binary notation of R R=6to61	
	61	0	1111111011101		
Output 4	0	23	11111111000101	11s	15+1
	0	24	11111111000110	00s	
	0	A	1111111	Binary notation of A A=23to255 s	
	0	255	1111111111111	11s	

Positive when s = 0
Negative when s = 1

EOB: end of code word group

Fig. 29

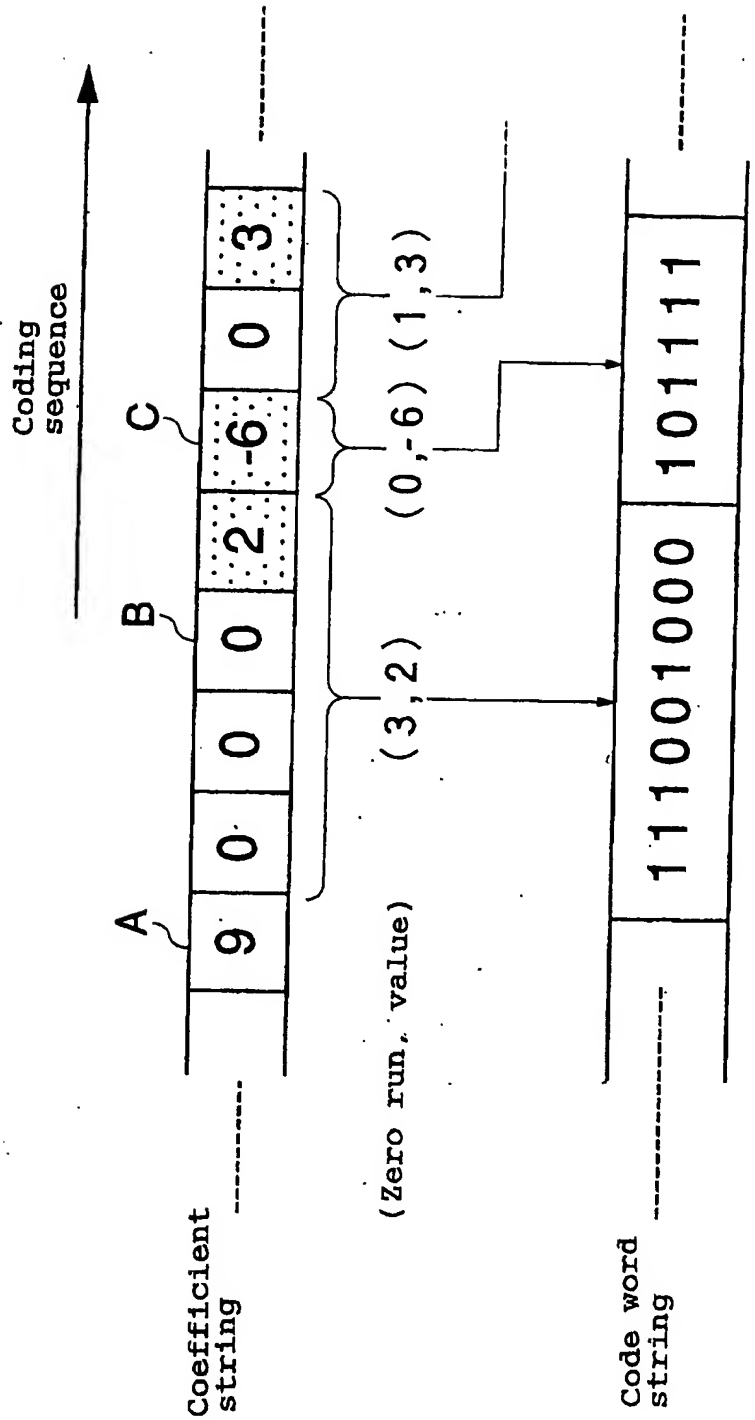

Coding sequence
A
B
C
9
0
0
0
2
-6
0
3
Coefficient string
(Zero run, value)
(3,2)
(0,-6)
(1,3)
Code word string
11100
1000
1011
111

Fig. 30

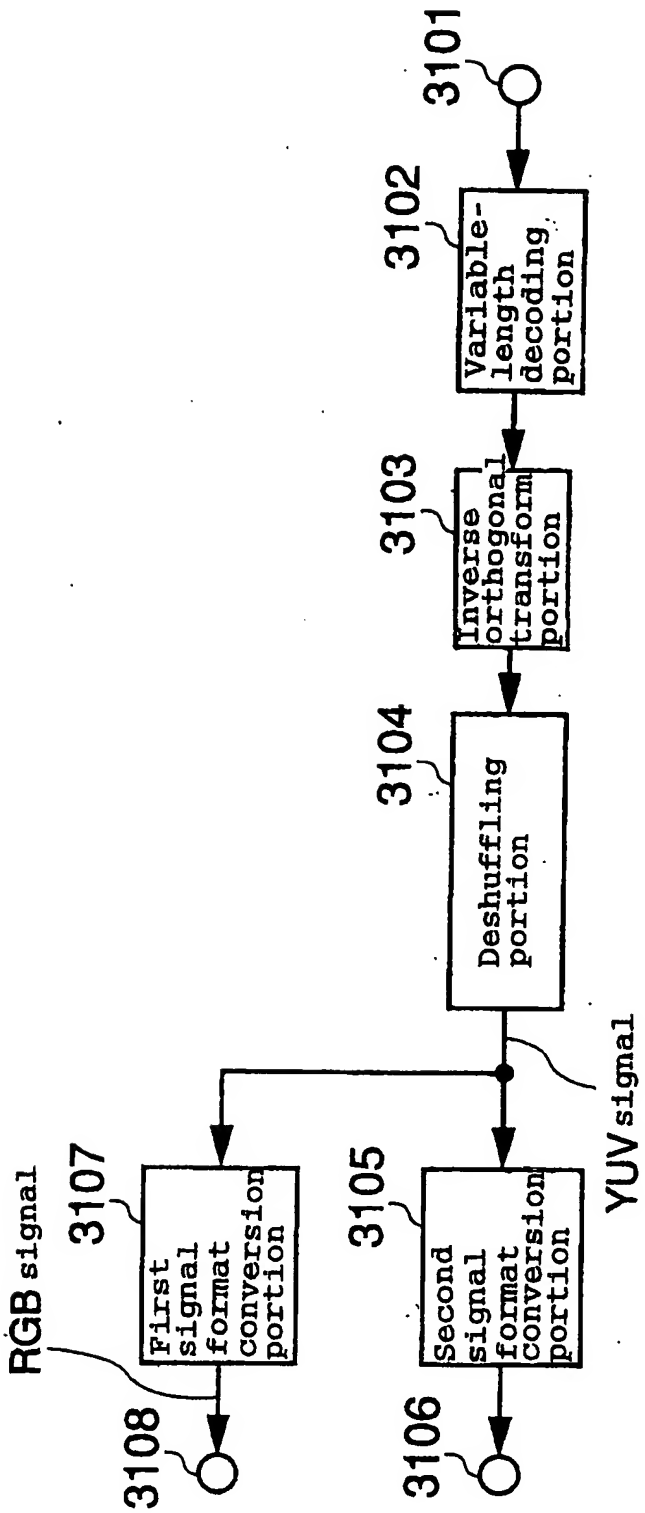
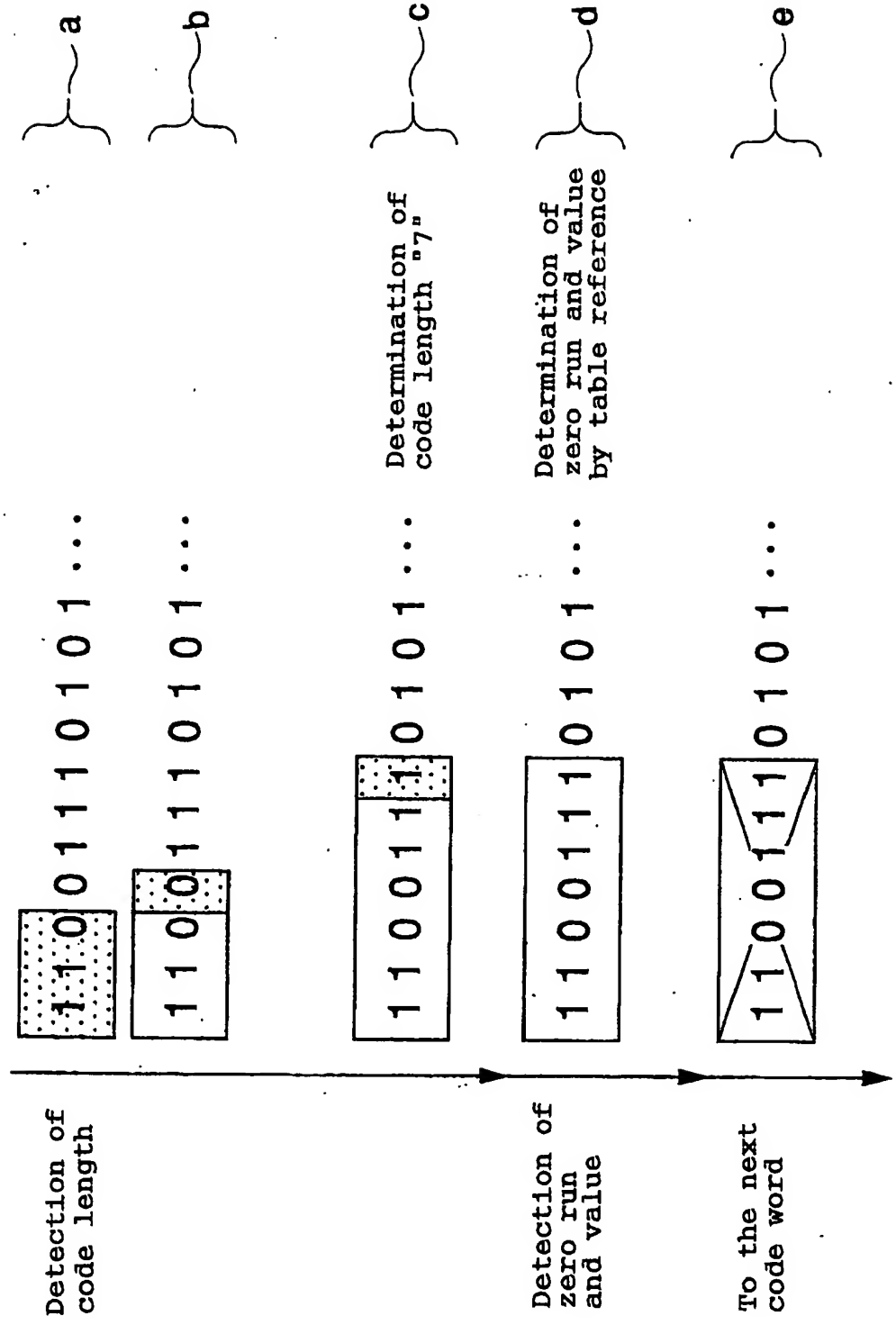

3101
3102
Variable-length decoding portion
3103
Inverse orthogonal transform portion
3104
Deshuffling portion
YUV signal
3105
Second signal format conversion portion
3106
3107
First signal format conversion portion
RGB signal
3108

Fig. 31

Fig. 32


a
b
c
d
e
1 1 0 0 1 1 1 0 1 0 1 ...
1 1 0 0 1 1 1 0 1 0 1 ...
1 1 0 0 1 1 1 0 1 0 1 ...
1 1 0 0 1 1 1 0 1 0 1 ...
1 1 0 0 1 1 1 0 1 0 1 ...
Determination of code length "7"
Determination of zero run and value by table reference
Detection of code length
Detection of zero run and value
To the next code word

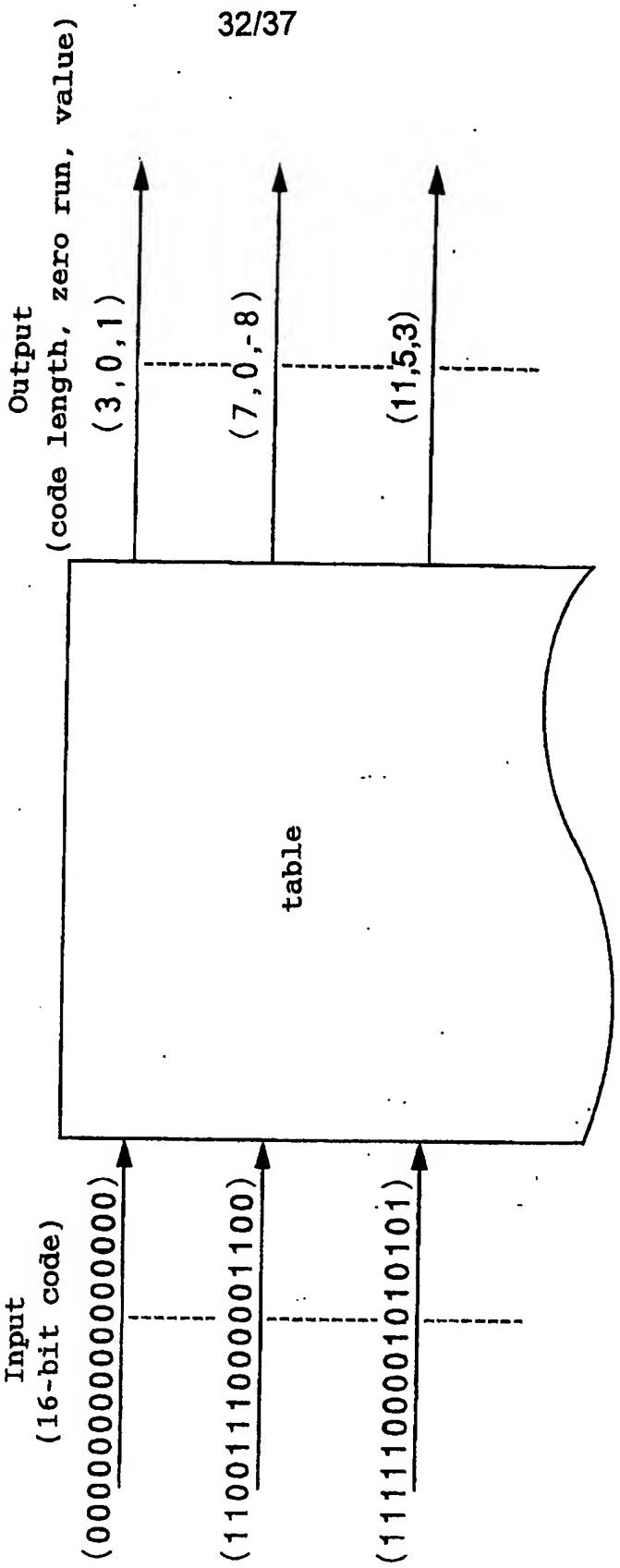

Output
(code length, zero run, value)
(3,0,1)
(7,0,-8)
(11,5,3)
table
Input
(16-bit code)
(0000000000000000)
(1100111000001100)
(1111100001010101)

Fig. 33

Fig. 34

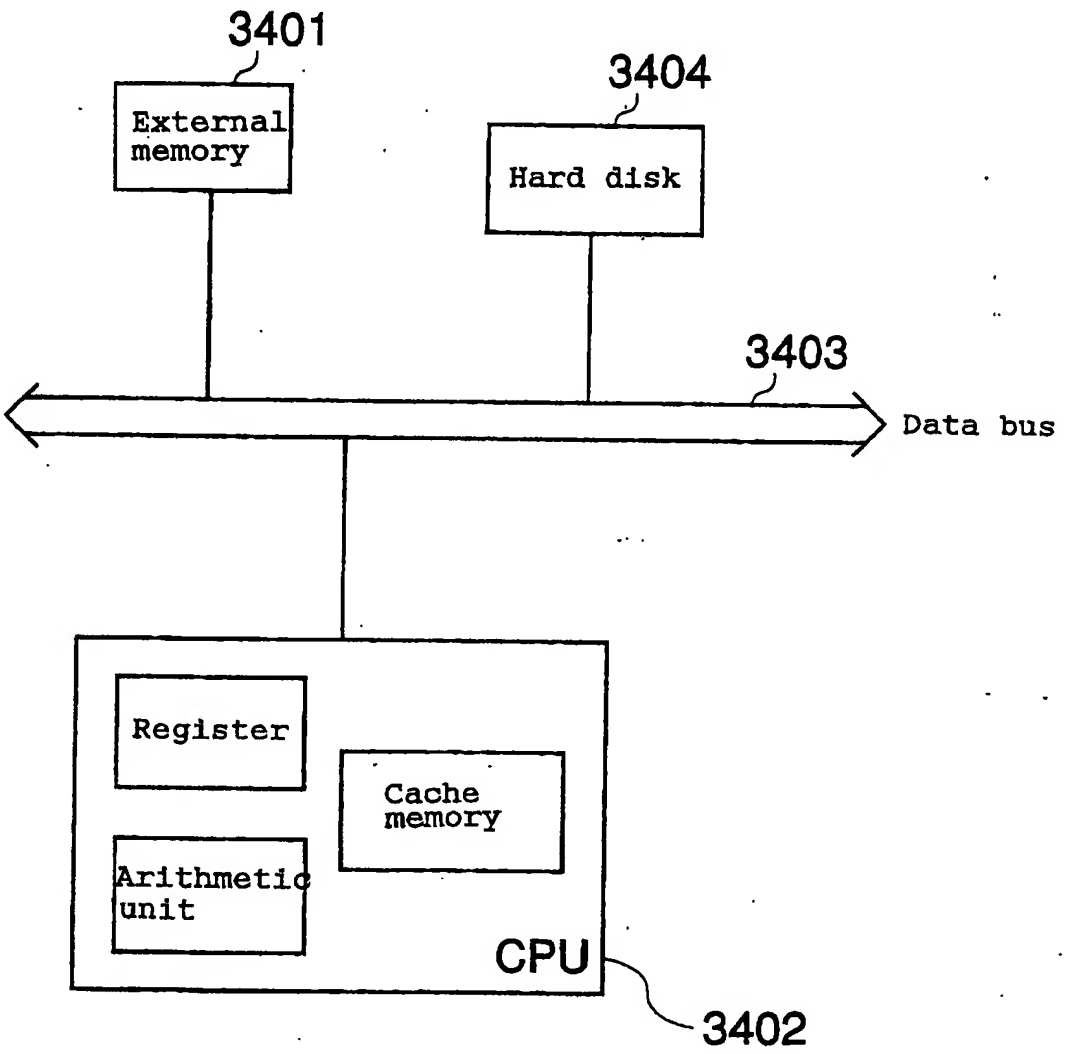

3401
External memory
3404
Hard disk
3403
Data bus
Register
Cache memory
Arithmetic unit
CPU
3402

Fig. 35

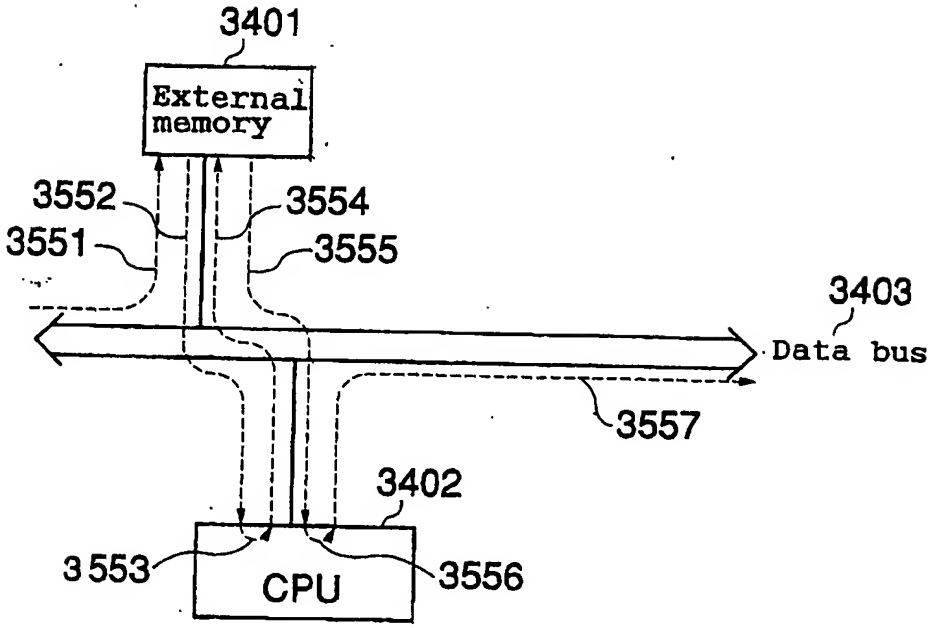
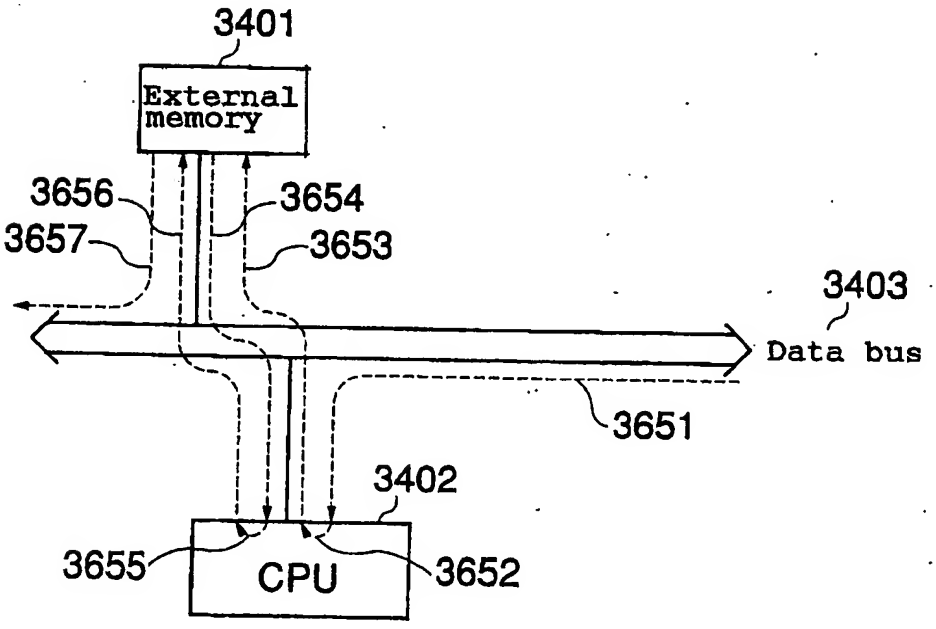

3401
External memory
3552
3554
3551
3555
3403
Data bus
3557
3402
3553
CPU
3556

Fig. 36


3401
External memory
3656
3654
3657
3653
3403
Data bus
3651
3402
3655
CPU
3652

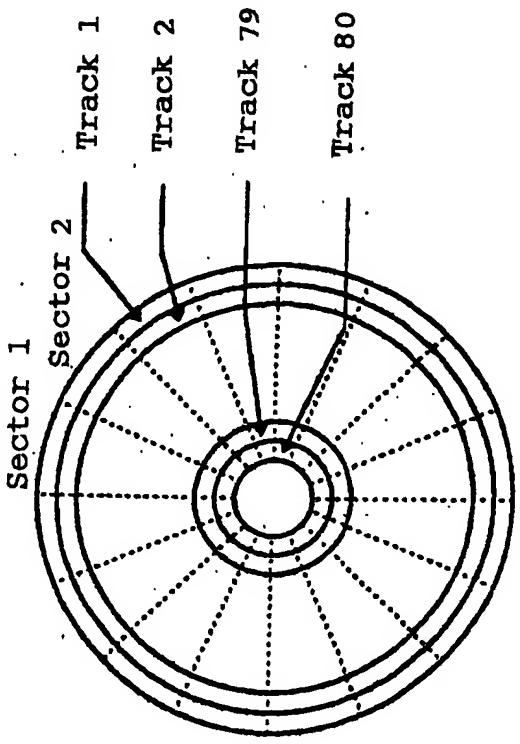


Fig. 37(a)

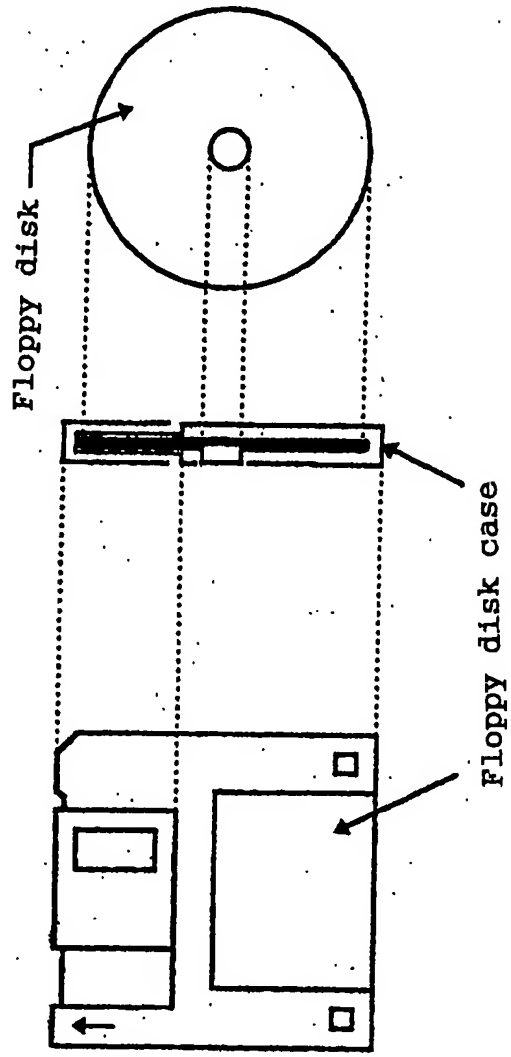


Fig. 37(b)

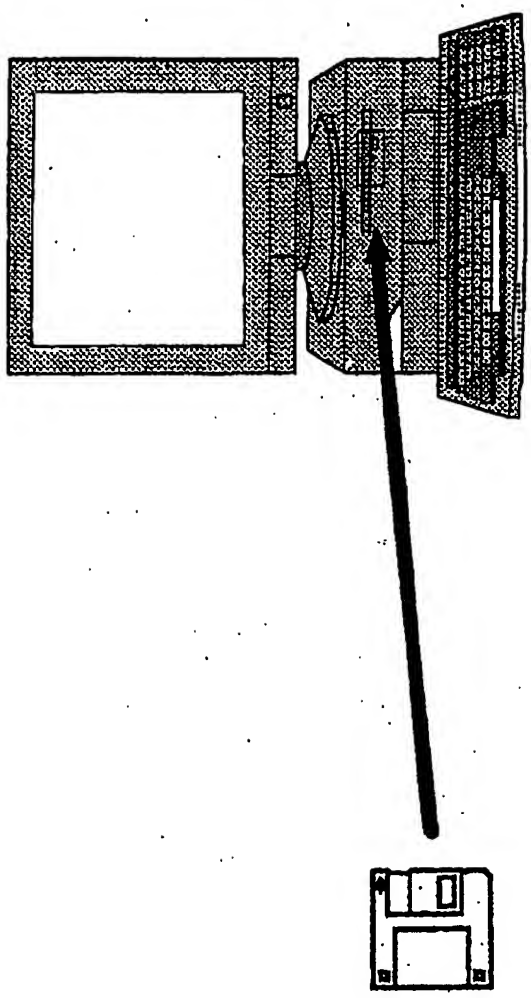


Fig. 37(c)